



STATISTICAL METHODS
FOR
IMAGE REGISTRATION AND DENOISING

DISSERTATION
Matthew D. Sambora
Lieutenant Colonel, USAF

AFIT/DCE/ENG/08-14

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

The views and conclusions contained in this dissertation are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, United States Government.

AFIT/DCE/ENG/08-14

STATISTICAL METHODS
FOR
IMAGE REGISTRATION AND DENOISING

DISSERTATION

Presented to the Faculty
Graduate School of Engineering and Management
Air Force Institute of Technology
Air University
Air Education and Training Command
In Partial Fulfillment of the Requirements for the
Degree of Doctor of Philosophy

Matthew D. Sambora, B.S.E.E., M.S.E.E.
Lieutenant Colonel, USAF

June 2008

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

STATISTICAL METHODS
FOR
IMAGE REGISTRATION AND DENOISING
Matthew D. Sambora, B.S.E.E., M.S.E.E.
Lieutenant Colonel, USAF

Approved:

Richard K. Martin

Richard Martin, PhD
(Chairman)

9 Jun 2008

Date

Stephen C. Cain

Stephen Cain, PhD
(Member)

6 June 08

Date

Matthew E. Fickus

Matthew Fickus, PhD
(Member)

9 Jun 08

Date

J. W. Chrissis

James Chrissis, PhD
(Dean's Representative)

6 Jun 08

Date

Accepted:

Marlin U. Thomas

Marlin U. Thomas, PhD
Dean, Graduate School of Engineering and Management

9 Jun 08

Date

Abstract

This dissertation describes research into image processing techniques that enhance military operational and support activities. The research extends existing work on image registration by introducing a novel method that exploits local correlations to improve the performance of projection-based image registration algorithms. The algorithm is shown to operate in low signal-to-noise ratio (SNR) conditions and to significantly improve registration performance by as much as a factor of 5.5 in mean-squared error over existing projection-based registration algorithms at a minimal computational cost.

The dissertation also extends the bounds on image registration performance for both projection-based and full-frame image registration algorithms and extends the Barankin Bound from the one-dimensional case to the problem of two-dimensional image registration. The Cramer-Rao and Barankin bounds are calculated for registration performed using 2-D registration algorithms and compared to bounds on registration estimates calculated using computationally efficient projection-based registration algorithms. It is demonstrated that in some instances, the Cramer-Rao lower bound is an overly-optimistic predictor of image registration performance and that under some conditions the Barankin bound is a better predictor of shift estimator performance. These conditions include low-SNR imaging and imaging under defocus error, two conditions which are frequently encountered in military imaging systems that employ passive infrared, light radar (LIDAR), and synthetic aperture radar (SAR).

The research looks at the related problem of single-frame image denoising using block-based methods. The research introduces three new algorithms for single-frame image denoising that operate by identifying regions of interest within a noise-corrupted image and then generating noise free estimates of the regions as averages

of similar regions in the image. The algorithms are shown to outperform Wiener and median filtering over a wide range of noise conditions but are most effective in images with very low signal-to-noise ratios.

Acknowledgements

When I worked in the Pentagon and someone asked me how I was doing, I would often joke, “I’m living my dream.” Here at AFIT, I was able to say that without irony. I am grateful to a number of people helped me get here and provided me with immeasurable support and assistance in the completion of this dissertation. First of all, I would like to thank my wife and daughter who gave up so much of their family time so that I could have this experience here at AFIT. I would also like to thank my advisor Dr. Richard Martin, who spent so much time reading and rereading my work. My committee members, Dr. Stephen Cain and Dr. Matthew Fickus, also gave their time freely and provided guidance, comments, and patient explanations. The Dean’s Representative on the committee, Dr. Chrissis, also provided immeasurable help with the style and layout of this document. I was also fortunate enough to have fellow students and members of the faculty and staff who were able to put a critical eye on my work and help me balance my family, military, and academic responsibilities. CDR Rob Stevens was someone I could always count on when I was the Senior Student Leader. Finally, I am grateful for the continuing support and mentorship of Brig Gen (ret.) Bradley Butler.

Matthew D. Sambora

Table of Contents

	Page
Abstract	iv
Acknowledgements	vi
List of Figures	x
List of Symbols	xiv
List of Abbreviations	xvi
 I. Introduction	 1
1.1 Problem Setting	1
1.1.1 Image Registration	2
1.1.2 Single-Frame Image Denoising	5
1.2 Purpose of the Research	7
1.3 Overview	7
 II. Literature Review	 9
2.1 Projection-Based Image Registration	9
2.2 Filtering Images to Facilitate Image Registration	12
2.3 Bounds on the Mean-Squared Error of Estimators	14
2.4 Calculating the Optical Transfer Function of an Imaging System	16
2.5 Existing Single-Frame Denoising Algorithms	17
2.5.1 Total Variation Minimization	18
2.5.2 Anisotropic Diffusion	20
2.5.3 Bilateral Filtering	21
2.5.4 Nonlocal Means	22
2.5.5 Patch-Based Denoising with Optimal Spatial Adaptation	24
2.5.6 Other Patch-Based Methods	25
2.6 Chapter Summary	25
 III. Improving Projection-Based Image Registration	 27
3.1 Improved Projection-Based Algorithm	28
3.1.1 Introduction of the Revised Figure of Merit	29
3.1.2 Use of the FOM in Filter Design	30
3.2 Observing Covariance in Images	32

	Page
3.3	Experimental Results 33
3.4	Chapter Summary 40
IV.	Bounds on Image Registration Algorithms 46
4.1	Performance Bounds on Image Registration With Filtered Projections 46
4.1.1	The CRLB of Registration Using Image Projection 48
4.1.2	The Barankin Bound on Registration Using Projections 48
4.2	Bounds of Two Dimensional Image Registration With Filtered Images 50
4.2.1	2-D CRLB with Optical Filtering 50
4.2.2	2-D Barankin Bound 51
4.3	Experimental Results 53
4.3.1	Registration Performance for Standard Pentagon Image 53
4.3.2	Registration Performance of Actual LIDAR data 56
4.4	Chapter Summary 59
V.	Block-based Methods for Denoising Images 62
5.1	The Gaussian Detection Denoising Method 62
5.1.1	Overview of the GDD Denoising Method 63
5.1.2	GDD Preliminary Assumptions and Calculations 64
5.1.3	Observed Distribution of Mean Squared Errors 66
5.1.4	Experimental Results with the GDD Algorithm 69
5.1.5	Conclusions Drawn from Initial Results 72
5.2	The HOD and XCD Denoising Algorithms 73
5.2.1	Higher-Order Statistics Method for Block Matching 75
5.2.2	Correlation-Based Method for Block Matching 83
5.2.3	HOD and XCD Simulation Results 85
5.3	Chapter Summary 91
VI.	Conclusions 97
6.1	Summary of Results and Contributions 97
6.1.1	Review of Results in Chapter III 97
6.1.2	Review of Results in Chapter IV 98
6.1.3	Review of Results in Chapter V 99
6.2	Recommended Future Research 101
6.2.1	Image Registration 101

	Page
6.2.2 Bounds on Registration Performance	102
6.2.3 Block-Based Denoising	103
Appendix A. Important Derivations	105
A.1 Calculation of the FOM Used in Chapter III	105
A.2 Derivation of Theoretical Performance Bounds	107
A.2.1 Derivation of the CRLB for a Projected & Fil- tered Image	107
A.2.2 Derivation of the Two-Dimensional CRLBs . .	110
Appendix B. Mathematical Background and Related Theory	114
B.1 The Chi-Square Distribution	114
B.1.1 Occurrence of the Chi-Square Distribution in Image Processing Problems	118
B.1.2 Statistical Characteristics of an Experimentally Determined Distribution	120
B.1.3 Statistics of the Sample Mean of $n \times n$ Noise Samples	121
B.2 Calculating the Covariance Present in Images and Image Projections	127
Bibliography	131

List of Figures

Figure		Page
1.1.	Demonstration of multiframe denoising.	3
1.2.	Examples of apparent redundant subimages in an image. . . .	6
3.1.	1024×1024 grayscale image of the Pentagon.	32
3.2.	Measured covariance of the column projections of the Pentagon image in Figure 3.1	32
3.3.	512×512 Brodatz grass image.	34
3.4.	Graph of measured covariance of column projections of the grass image in Figure 3.3 calculated without noise and again with AWGN of $\sigma = 100$ (PSNR = 8.12).	34
3.5.	1024×1024 Brodatz sand image.	35
3.6.	Graph of measured covariance of column projections of the sand image in Figure 3.5 calculated without noise and again with AWGN of $\sigma = 100$ (PSNR = 8.13).	35
3.7.	1024×1024 Brodatz water image.	36
3.8.	Graph of measured covariance of column projections of the image in Figure 3.7 calculated without noise and again with AWGN of $\sigma = 100$ (PSNR = 8.13).	36
3.9.	Calculated F_{P_y} , and MSE for the Pentagon image in Figure 4.1 with $\sigma_{noise} = 100$, actual shift = 0.	42
3.10.	Calculated F_{P_y} , and MSE for the grass image in Figure 3.3 calculated without noise and again with AWGN of $\sigma = 100$. . .	42
3.11.	Calculated F_{P_y} , and MSE for the sand image in Figure 3.5 with $\sigma_{noise} = 100$, actual shift = 0.	42
3.12.	Calculated F_{P_y} , and MSE for the water image in Figure 3.7 with $\sigma_{noise} = 100$, actual shift = 0.	42
3.13.	Optimal spatial-domain filtering kernel for the Pentagon image in Figure 4.1.	43
3.14.	Optimal spatial-domain filtering kernel for the Brodatz water image in Figure 3.7.	43
3.15.	512×512 Tank image from http://sipi.usc.edu/database/ . .	44
3.16.	Calculated covariance function for the 512×512 Tank image.	44
3.17.	Calculated F_{P_y} , and MSE for the tank image in Figure 3.15 with AWGN of $\sigma = 100$	44
3.18.	Aerial image of cornfield from a sequential series of frames. .	45
3.19.	Graph of kernel vs MSE for the image of Figure 3.18 with AWGN of $\sigma = 20$ when measuring an actual shift of zero. . .	45
3.20.	An aerial image of a road taken from the series.	45

Figure		Page
3.21.	Comparison of the calculated F_{P_y} , for the images in Figs. 3.18 and 3.20.	45
4.1.	1024×1024 Pentagon image from http://sipi.usc.edu/database/	54
4.2.	Bound on the variance of estimates of the x -shift for the image shown in Figure 4.1 using both projections and 2-D registration.	54
4.3.	128×128 subsection of Figure 4.1.	55
4.4.	Calculated performance bounds for registration using projections and 2-D registration for the 128×128 subsection of the Pentagon image.	55
4.5.	Image shown in Figure 4.1 with simulated 0.7λ defocus error.	56
4.6.	Bounds for the image shown in Figure 4.1 with simulated 0, 0.1λ , 0.3λ , and 0.7λ defocus errors.	56
4.7.	Image shown in Figure 4.3 with simulated 0.7λ defocus error.	57
4.8.	Bounds for the image shown in Figure 4.3 with simulated 0, 0.1λ , 0.3λ , and 0.7λ defocus errors.	57
4.9.	256×256 image resulting from median filtering, and averaging 50 frames of LIDAR data captured at 10 km from the target.	59
4.10.	Representative LIDAR frame prior to filtering and averaging (PSNR = 26.3).	59
4.11.	68×168 region of interest within the image of Figure 4.9.	59
4.12.	Representative region of interest in a LIDAR frame prior to filtering and averaging (PSNR = 24.6).	59
4.13.	Bounds on registration using projections and 2-D registration of the LIDAR frame shown in Figure 4.9.	60
4.14.	Bounds on registration using projections of the LIDAR frame region of interest shown in Figure 4.11.	60
5.1.	Graph of the PDF $\chi_{225}^{\prime 2}(0)$ over $0 \leq x \leq 500$	67
5.2.	Histogram of the Mean Squared Error between a representative $\hat{\mathbf{F}}_{i,j}$ and all $\hat{\mathbf{G}}_{s,t}$	68
5.3.	Histogram resulting from the calculation of $\ \hat{\mathbf{F}}_{i,j} - \mathbf{A}(s, t)\hat{\mathbf{G}}_{s,t}\ _2^2$ for a single $\hat{\mathbf{F}}_{i,j}$ and all values of $\mathbf{A}(s, t)\hat{\mathbf{G}}_{s,t}$	70
5.4.	Truth image used to generate the simulation data.	71
5.5.	Truth image with AWGN of $\sigma = 25$ added.	72
5.6.	Histogram of the Noisy Image.	73
5.7.	Output obtained using the NLM means algorithm.	74
5.8.	Output obtained using the GDD method.	75
5.9.	Output obtained by GDD denoising using 49 of 10201 possible 15×15 blocks.	76
5.10.	Output obtained by GDD denoising using an overlapping lattice of 98 of 10201 possible 15×15 blocks.	77

Figure		Page
5.11.	Plot of the measured maximum skewness vs. N - the square root of the block size.	81
5.12.	Graph of results comparing output of HOD filtering, XCD filtering, Wiener filtering and median filtering.	86
5.13.	Block size vs. HOD output for a 101×101 image.	87
5.14.	Block size vs. XCD output for the 101×101 image.	87
5.15.	512×512 image of a tank derived from LIDAR data.	89
5.16.	Image of Figure 5.15 corrupted with noise of $\sigma = 9000$, input PSNR = 18.96.	89
5.17.	Image in Figure 5.16 denoised using HOD and a block size of six.	89
5.18.	Image of Figure 5.16 denoised using Wiener filtering. Output PSNR = 26.23.	89
5.19.	Original 256×256 LIDAR image of a tank resulting from a multiframe average.	94
5.20.	Tank image with Additive White Gaussian Noise, $\sigma_{noise} = 5000$, input PSNR = 19.33.	94
5.21.	Tank image with Additive White Gaussian Noise, (input PSNR = 19.33) after Wiener filtering. Output PSNR = 26.43.	94
5.22.	Tank image with Additive White Gaussian Noise, (input PSNR = 19.33) after HOD filtering with block size of 5, output PSNR = 31.00.	94
5.23.	The “method noise” derived by subtracting the denoised image found in Figure 5.17 from the original noisy image found in Figure 5.16 Figure 5.16.	95
5.24.	Results of the HOD and XCD methods compared against the Exemplar-based image denoising algorithm.	95
5.25.	Results of the HOD and XCD methods compared against the Bayeshrink and SUREshrink algorithms.	96
B.1.	The PDF of $\chi_{\nu}^{\prime 2}(\lambda)$ with $\lambda = 0$	116
B.2.	The PDF of $\chi_{\nu}^{\prime 2}(\lambda)$ with $\lambda = 1$	116
B.3.	The PDF of $\chi_{\nu}^{\prime 2}(\lambda)$ with $\lambda = 3$	117
B.4.	The PDF of $\chi_{\nu}^{\prime 2}(\lambda)$ with $\lambda = 6$	117
B.5.	Graph of the PDF of $\chi_{\nu}^{\prime 2}(\lambda)$ with $\lambda = 10$	118
B.6.	Analytically constructed plot of the PDF of the mean of an ensemble of noise samples.	122
B.7.	Experimentally constructed plot of the PDF of the mean of an ensemble of noise samples.	123
B.8.	Per pixel error between similarly constructed images.	125
B.9.	Comparison of measured vs. predicted data with $\chi_{\nu}^{\prime 2}(\lambda)$ with $\lambda = 15$ distribution.	128

Figure		Page
B.10.	1024 \times 1024 grayscale image of the Pentagon.	129
B.11.	Measured covariance of the column projections of the image in Figure 4.1.	129
B.12.	256 \times 256 grayscale aerial image of a chemical plant.	130
B.13.	Measured covariance of the column projections of the image in Figure B.12.	130
B.14.	Measured covariance of column projections of the pentagon image with AWGN of $\sigma = 100$	130
B.15.	Measured covariance of the column projections of the image in Figure B.12 with AWGN of $\sigma = 100$	130

List of Symbols

Symbol		Page
\mathbf{T}_α	One-dimensional translational shift operator	3
$\mathbf{T}_{\alpha,\beta}$	Two-dimensional translational shift operator	3
$\mathbf{d}_{i,x}(y)$	Point on on a vertical projection of a frame of data . . .	4
$\mathbf{d}_{n,y}(x)$	Point on on a horizontal projection of a frame of data .	4
$\mathbf{i}_y(x)$	Projection of a diffraction-limited image	5
$\mathbf{q}_n(x)$	Projection of the noise component of an image	5
\mathbf{W}_f	Windowing operator for a vector of data	10
m_p	Index of the center point of a projection	11
δ_s	Maximum allowable shift value between two frames of data	11
\mathbf{J}	Fisher Information Matrix	14
Θ	Vector of non-random parameters	14
$L(d \Theta_i, \Theta_0)$	Likelihood function used to calculate the Barankin Bound	15
\mathcal{H}	Optical transfer function of an imaging system	17
\mathbf{H}_o	Inverse Fourier transform of the optical transfer function	17
div	The divergence of a gradient	20
∇	Gradient	20
\mathbf{A}	Matrix of weighting factors	23
\mathcal{I}	The identity matrix	54
$\langle \mathbf{F}, \mathbf{G} \rangle$	The inner product of matrices \mathbf{F} and \mathbf{G}	66
$\Re\{\}$	Real part of a complex number	66
x	A random variable with a $\chi_\nu'^2(\lambda)$ distribution	114
ν	Degrees of freedom of a $\chi_\nu'^2(\lambda)$ distribution	115
λ	noncentrality parameter of a $\chi_\nu'^2(\lambda)$ distribution	115
$\Gamma(u)$	Gamma function	115
$\text{VAR}[x]$	Variance of x	115
$\chi_\nu'^2(\lambda)$	Noncentral chi-square distribution	115
$\Delta_{s,t}$	The error between a given subimage and one centered at s, t	119
σ	Standard deviation of a random variable	119
$\ \mathbf{A}\ _F$	The Frobenius norm of matrix \mathbf{A}	120
$\langle \rangle$	Sample mean	121
$\mathbf{F}_{i,j}$	Mean-subtracted subimage to be denoised	124
$\mathbf{G}_{s,t}$	Statistically similar, zero-mean blocks used to denoise $\mathbf{F}_{i,j}$	124

Symbol		Page
$\hat{\Delta}_{s,t}$	Estimated error between $\mathbf{F}_{i,j}$ and $\mathbf{G}_{s,t}$	124
$\mathbb{E}[x]$	Expected value of x	126

List of Abbreviations

Abbreviation		Page
SNR	Signal-to-Noise ratio	7
CCD	Charge-Coupled Device	9
FFT	Fast Fourier Transform	10
FPN	Fixed Pattern Noise	10
FOM	Figure of Merit	11
PSD	Power Spectral Density	13
CRLB	Cramer-Rao lower bound	14
OTF	Optical Transfer Function	16
ATF	Amplitude Transfer Function	16
TV	Total Variation	18
NLM	Nonlocal Means	22
OSA	Optimal Spatial Adaptation	24
i.i.d.	Independent and Identically Distributed	30
PSNR	Peak Signal-to-Noise Ratio	37
PDF	Probability-Distribution Function	47
FIM	Fisher Information Matrix	48
GDD	Gaussian Detection Denoising	63
LIDAR	Light-Radar	69
HOD	Higher-Order Denoising algorithm	75
XCD	Cross Correlation Denoising algorithm	83
ATR	Automatic Target Recognition	102

STATISTICAL METHODS FOR IMAGE REGISTRATION AND DENOISING

I. Introduction

For the United States Air Force, images are crucial for real-time intelligence, target planning, and flight operations. Images are also analyzed extensively by military personnel engaged in medical diagnostics, nondestructive inspection, astronomy, security, law enforcement and counterintelligence. Not only are these images used for a variety of purposes, they are also taken under a variety of adverse conditions and with illumination sources that range in frequency from ultrasound to x-ray. In order to minimize power consumption to avoid detection of the receiver in an operational scenario, the imaging systems being used may be passive (as with passive infrared systems) or may use signals of opportunity such as street lights for illumination sources. Compounding these challenges, the receiver may be located at a distance miles away from the target, may have only a fleeting glance at a target, and may be degraded by severe thermal or weather conditions. These systems may also be used in unpredictable environments with insufficient illumination, significant background noise, and spurious electronic emissions. Despite these challenges, as noted by Driggers *et al.* [20], these imaging systems may be used to determine whether or not to fire on a particular target. Thus, the ability to receive and interpret images is key to making decisions having lethal consequences.

1.1 Problem Setting

The ongoing challenge addressed by the research described in this dissertation is the estimation of underlying image parameters from available data when that data

is corrupted by additive noise. Some amount of noise is always present in digital imaging systems for reasons eloquently described by Snyder *et al.* [51]. These noise sources include random variations in photon conversions during object illumination, thermal noise in the sensor, background noise caused by luminescent radiation in the area of the sensor, sensor biases, and random noise induced by the sensor amplifier during readout of the individual pixel.

The approach used in this research to solve this estimation problem is to fuse available information to estimate the noise-free intensity of a scene, the motion of the imaging sensor, or the motion of an object within an image. One aspect of this problem that will be explored is the fusion of multiple frames of the same scene to create an estimate of intensity scene that is better than any single frame of that scene.

1.1.1 Image Registration. When a sensor is able to take many frames of the same scene, or when many frames of the same scene are available from different sensors, it is possible to combine these frames to estimate the parameters of the scene or the sensors. A simple example of this is shown in Figure 1.1 where several frames of the same scene are averaged to form an improved estimate of an image of a tank. However, the ability to perform this fusion is much more complicated than this figure indicates.

In an actual imaging system the sensor creating the image may be moving or vibrating and the frames of the image captured by the sensor may be separated by some unknown spatial offset. The process of estimating this offset and aligning the image frames is called image registration [7]. If the images can be registered correctly, the estimated offset of the frames may be used to estimate the motion of the sensor. This real-time application of the data requires registration algorithms that are fast and highly accurate.

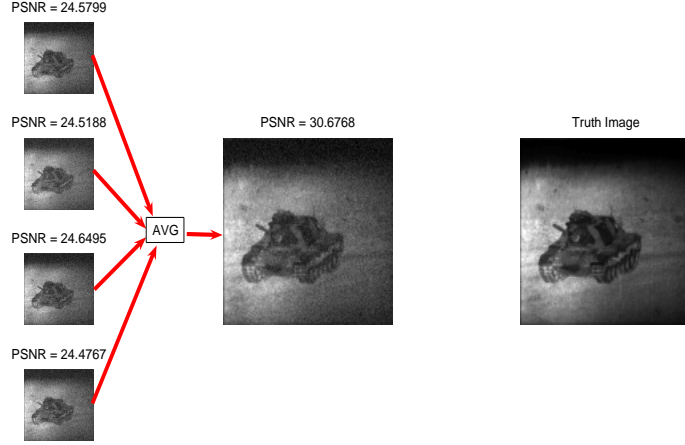


Figure 1.1: Example of multiple frames of the same scene used to improve the estimate of a scene.

The research proposed in this dissertation addresses this need by introducing a novel method for optimizing the performance of projection-based image registration algorithms. To keep this examination tractable, the problem is bounded to include only translational shifts. The mathematical model which will be used throughout this dissertation to describe this operation is now defined where it is assumed there are two $N \times N$ observations of an image \mathbf{I} corrupted with additive white Gaussian noise (AWGN) \mathbf{Q} where $\mathbf{I}(x, y), \mathbf{Q}(x, y) \in \mathbb{R}$ so that $\mathbf{D}(x, y)^* = \mathbf{D}(x, y)$ under complex conjugation.

For the one-dimensional case, an operator \mathbf{T}_α is defined that acts on a vector \mathbf{i} such that

$$(\mathbf{T}_\alpha \mathbf{i})(x) \equiv \mathbf{i}(x - \alpha) \quad (1.1)$$

For the two-dimensional case, an operator $\mathbf{T}_{\alpha, \beta}$ is defined that acts on a vectorized version of \mathbf{I} such that

$$(\mathbf{T}_{\alpha, \beta} \mathbf{I})(x, y) \equiv \mathbf{I}(x - \alpha, y - \beta) \quad (1.2)$$

\mathbf{T}_α and $\mathbf{T}_{\alpha,\beta}$ are unitary, norm-preserving operators that can be realized using circulant matrices and have the properties

$$\mathbf{T}_{\alpha_1} \mathbf{T}_{\alpha_2} = \mathbf{T}_{(\alpha_1+\alpha_2)}, \quad (1.3)$$

$$(\mathbf{T}_\alpha)^T = \mathbf{T}_{-\alpha}, \quad (1.4)$$

$$\mathbf{T}_{\alpha_1,\beta_1} \mathbf{T}_{\alpha_2,\beta_2} = \mathbf{T}_{\alpha_1,(\beta_1+\beta_2)}, \quad (1.5)$$

$$\mathbf{T}_{\alpha_1,\beta_1} \mathbf{T}_{\alpha_2,\beta_1} = \mathbf{T}_{(\alpha_1+\alpha_2),\beta_1}, \quad (1.6)$$

$$(\mathbf{T}_{\alpha,\beta})^T = \mathbf{T}_{-\alpha,-\beta}. \quad (1.7)$$

Subscripting with $n \in \{1, 2\}$ to indicate the number of the observation, two frames of image data are now defined

$$\mathbf{D}_1 = \mathbf{I} + \mathbf{Q}_1, \quad (1.8)$$

$$\mathbf{D}_2 = \mathbf{T}_{\alpha,\beta} \mathbf{I} + \mathbf{Q}_2, \quad (1.9)$$

where α and β are shifts of the diffraction-limited image \mathbf{I} in the x and y directions respectively. Furthermore, the additive noise, \mathbf{Q}_1 and \mathbf{Q}_2 are defined to be independent and identically distributed Gaussian with zero mean and variance σ^2 .

The optimization method introduced by this dissertation employs projections of sequential image frames to create an estimate of their spatial offset. This is shown to be a computationally simple approach that is also highly accurate. The x and y projections of \mathbf{D}_n are defined in the dissertation as $\mathbf{d}_{i,x}(y)$ and $\mathbf{d}_{n,y}(x)$ where [11]

$$\mathbf{d}_{n,x}(y) = \sum_{x=0}^{N-1} \mathbf{D}_n(x, y), \quad (1.10)$$

$$\mathbf{d}_{n,y}(x) = \sum_{y=0}^{N-1} \mathbf{D}_n(x, y). \quad (1.11)$$

Expanding these terms for frame i in direction y yields

$$\mathbf{d}_{n,y}(x) = \underbrace{\sum_{y=0}^{N-1} \mathbf{I}(x, y)}_{\mathbf{i}_y(x)} + \underbrace{\sum_{y=0}^{N-1} \mathbf{Q}_n(x, y)}_{\mathbf{q}_n(x)}, \quad (1.12)$$

where $\mathbf{i}_y(x)$ and $\mathbf{q}_n(x)$ are introduced as notation for the projection of the diffraction limited image and the projection of the noise in the image. Note that $n \in \{1, 2\}$ is an index and \mathbf{i} is a vector projection of the data. The research described in the dissertation includes the introduction of a novel method for filtering these projections to improve the accuracy of estimates of image translations.

The accuracy of estimates on translations between images is a function of image content and the amount of additive noise in the images. Limits on this accuracy have been predicted using information-theoretic analytical tools such as the Cramer-Rao lower bound [53]. However, this commonly-used bound provides an incomplete description of the behavior exhibited by image registration algorithms under the high-noise conditions present in some military imaging systems. Under high-noise conditions, the Cramer-Rao lower bound is shown in this research to be an overly-optimistic predictor of estimator performance.

To address this shortcoming, this dissertation applies the Barankin bound to the image registration problem. This bound, used traditionally for estimating translational shifts between one-dimensional signals [37, 38, 41], is extended here to the two-dimensional problem of image registration. The Barankin bound is used to find a lower bound on the mean-squared error of shift estimates generated from images that are out of focus and corrupted by noise. It is also employed to compare the performance of registration estimates generated from projections.

1.1.2 Single-Frame Image Denoising. Single-frame image denoising algorithms combine information from a single noise-corrupted image to estimate the

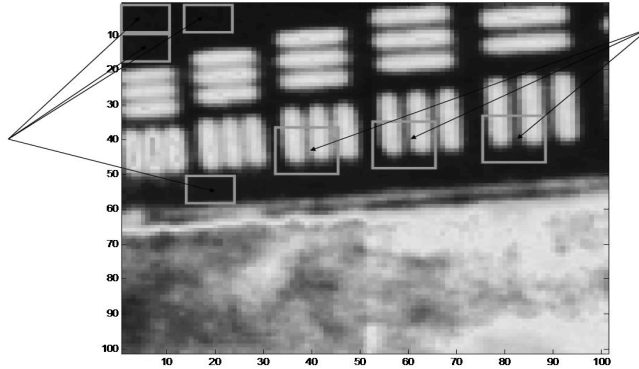


Figure 1.2: Examples of apparent redundant subimages in an image.

intensity of the underlying noise-free scene. A variety of approaches have been used to perform this denoising [1,6,9,12,13,16,17,31,32,40,49,52]. All of these approaches employ some measure of similarity for comparing and combining individual pixels in an image. Several of these methods [9,31,32] compare pixels by measuring the similarity of local neighborhoods around these images.

As shown in Figure 1.2, it is often a simple exercise to at least roughly identify similar regions within images that may contribute to a noise-reduced block average. If mean of the noise in the redundant subimages is zero, averaging should yield a result that is close, in an L2-norm sense, to the diffraction-limited subimage. This process is similar to multiframe image denoising using small frames but has the additional challenge of differentiating similar from dissimilar content. These methods have been shown to be effective [9,31,32]; however, the methods in the literature also rely on parameters that are specific to the image under study and may be further improved.

This dissertation introduces several new image processing algorithms that compare and combine subimages within a frame to provide denoising performance that

is on par with, and in some cases superior to, the performance achieved by other state-of-the-art algorithms. Unlike similar block-based algorithms, these algorithms suspend the requirement for similar blocks to be located in close spatial proximity to each other. This is shown to improve denoising performance over existing block-based methods in the most severely corrupted images (as measured by signal-to-noise ratio (SNR)) while reducing reliance on image-specific parameters in the algorithm.

1.2 Purpose of the Research

The research described in this dissertation examines the interrelated problems of image registration and image denoising. It develops novel methods to improve the performance of existing registration and denoising algorithms. It also extends the general understanding of the performance of image registration algorithms by extending the existing one-dimensional applications of the Barankin bound to the two-dimensional problem of image registration.

1.3 Overview

As a synopsis of the dissertation structure, the following outline is provided: Chapter II provides a review of related research and theoretical background that will support discussion of the concepts in the following chapters. Chapter III introduces a computationally-efficient method for estimating translational shifts of frames of the same scene. This method can be used for estimating the motion of a sensor between frames or to facilitate multiframe averaging. Chapter IV examines and extends the theoretical bounds on estimates of shifts between similar images and subimages. The effects of filtering, projecting and defocus errors in images are examined and the Barankin bound on image registration is introduced and demonstrated to be relevant to projections and low-intensity image processing. Chapter V introduces several new algorithms that exploit regularity in an image to facilitate denoising when only a single frame of a scene is available. The dissertation concludes in

Chapter VI with a summary of the research in this dissertation and proposals for further work on statistical image processing problems.

II. Literature Review

The problems of image registration and single-frame image denoising have been extensively studied in the literature. This survey of existing work looks at those methods from recent literature that offer the most computationally-efficient methods for estimating shifts between images. This review identifies opportunities for improving on existing methods for projection-based image registration and identifies a method for improving the accuracy of estimates on the performance limits of registration estimates. The review also examines statistical methods rooted in information theory for analyzing the performance of those methods. The review then covers a number of related single-frame image denoising algorithms that are similar in nature to the multiframe averaging methods that are facilitated by image registration. The review provides insight into ways that existing block-based algorithms may be improved.

2.1 Projection-Based Image Registration

Image registration as described in this dissertation is the process of spatially aligning images which may have been taken by different sensors or were captured at different times. This spatial alignment may be used for multiframe image denoising [36] or for estimating camera motion [46]. Images may be registered using a variety of techniques including cross correlations, Fourier transforms, and by identifying and aligning features within different images. An overview of these and other image registration algorithms is available in [7].

If an imaging sensor is used primarily for motion estimation, image projections offer what is perhaps the fastest approach for registering available images at a low computational cost [11, 45]. This speed is the result of faster data acquisition times and reduced computational complexity. As described by Cain [10], existing Charge-Coupled Devices (CCDs) need to be read out serially if an entire image is to be

acquired. However, if only a projection is required, the projection may be formed by integrating charges corresponding to photon counts across the vertical and horizontal axes of some existing CCDs [10]. Although less image information is available in the projections than in a full 2-D image, Cain *et al.* note that in the presence of fixed pattern and temporal noise, projection-based methods can provide performance that is actually superior to that of 2-D cross correlations [11].

In addition, registering the image projections requires two 1-D cross correlations instead of one 2-D cross correlation. Cross-correlation of $N \times N$ 2-D images is most efficiently performed in the frequency domain and requires a total of three fast Fourier transforms (FFTs) yielding a computational complexity of $\mathcal{O}(6N^2 \log N)$. For the same image, the cross-correlation of its projections in the frequency domain requires six FFTs for a total complexity of $\mathcal{O}(6N \log N)$. Using projections, methods involving only real numbers (and possibly only integers) become feasible yielding further reductions in computational complexity. For motion estimation purposes, this combination of readout speed and low computational complexity makes these algorithms especially attractive.

Cain *et al.* [11] describe how registering images using their projections can improve performance over 2-D correlation methods when significant fixed pattern noise (FPN) is present in the images. The mechanism behind this is that the signal in the projections is correlated and the FPN is assumed to be uncorrelated. They go on to note that the ability to correctly register two images in correlation-based image registration is dependant not only on the height of the autocorrelation peak, but also on the difference between the peak and other points on the autocorrelation.

A windowing operator is used in their calculation which is denoted as \mathbf{W}_f and is realized computationally as a diagonal matrix where the diagonal elements are

$$\mathbf{W}_f(z, z) = \begin{cases} 1 & : |z - m_p| \leq (N/2 - \delta_s) \\ 0 & : \text{else} \end{cases} \quad (2.1)$$

where the variable m_p is the value of z corresponding to the midpoint of a projection (i.e. the index of the center point of the projection) and δ_s is the maximum allowable shift value between the frames. This windowing function is necessary to ensure that the data overlaps for a sufficient number of terms of the cross-correlation and mitigates the reduction in cross correlation power that would otherwise occur for two identically sized projections.

Using the projections and the windowing function notation, Cain *et al.* compute the 1-D cross correlations of the windowed projections of two images as [11]

$$\mathbf{p}_y = \mathbf{d}_{1,y} * \mathbf{W}_f \mathbf{d}_{2,y} - \overline{\mathbf{W}_f \mathbf{d}_{1,y}} \odot \overline{\mathbf{d}_{2,y}}, \quad (2.2)$$

$$\mathbf{p}_x = \mathbf{d}_{1,x} * \mathbf{W}_f \mathbf{d}_{2,x} - \overline{\mathbf{W}_f \mathbf{d}_{1,x}} \odot \overline{\mathbf{d}_{2,x}}. \quad (2.3)$$

where $\overline{\mathbf{d}_{i,z}}$, $z \in \{x, y\}$ denotes a vector of length N with all elements equal to the scalar average of a given projection and $\overline{(\mathbf{W}_f \mathbf{d}_{1,z})}$ is a vector with the elements equal to the mean of $\mathbf{d}_{i,z}$ over a windowed area for a given index of $\mathbf{d}_{i,z}$. The notation $*$ is used to indicate convolution and \odot to indicate a Hadamard multiplication. The windowing function in (2.2) and (2.3) effectively bounds the search area for the registration peak to a subregion within $\mathbf{d}_{1,x}$ or $\mathbf{d}_{1,y}$. This windowing function should be defined based on a bound on the translation between frames. Large translations will necessitate a small window and vice versa.

The shift estimate is computed from the projections defined in (2.2) and (2.3) and is calculated as

$$\hat{\alpha} = \arg \max_z |\mathbf{p}_y(z)|, \quad (2.4)$$

$$\hat{\beta} = \arg \max_z |\mathbf{p}_x(z)|. \quad (2.5)$$

Cain *et al.* [11] also describe a figure of merit (FOM) that can be used to evaluate the ability to differentiate the shift between two images. Cain's FOM is

defined such that z is the integer-valued index of a point on the cross correlation between two projections and α and β are the actual shifts in the x and y directions. The FOM is stated

$$F_{P_y}(z, \alpha) \triangleq \frac{(\mathbb{E}[\mathbf{p}_y(\alpha)] - \mathbb{E}[\mathbf{p}_y(z)])^2}{\mathbb{E}[\text{VAR}[\mathbf{p}_y(z)|\mathbf{i}] + \text{VAR}[\mathbf{p}_y(\alpha)|\mathbf{i}]}, \quad (2.6)$$

$$F_{P_x}(z, \beta) \triangleq \frac{(\mathbb{E}[\mathbf{p}_y(\beta)] - \mathbb{E}[\mathbf{p}_y(z)])^2}{\mathbb{E}[\text{VAR}[\mathbf{p}_y(z)|\mathbf{i}] + \text{VAR}[\mathbf{p}_y(\beta)|\mathbf{i}]}, \quad (2.7)$$

where the notation $\mathbb{E}[\cdot]$ indicates the expected value of a random variable and z is used as an integer index for the cross correlation.

Although this FOM is accurate for the algorithm described, if the projections are filtered the noise in adjacent points of the cross correlation becomes correlated and the assumptions under which the FOM was developed no longer apply. This points to a need to further generalize the FOM to account for filtering.

2.2 Filtering Images to Facilitate Image Registration

Filtering images prior to their registration has been discussed extensively in the literature. It has also been used in the related problem of time-delay estimation of 1-D signals [33]; however, the bounds on registration from filtered images has not been fully explored.

Barron [3] notes that prefiltering to smooth images is a common first stage of many image registration algorithms. Filtering to improve registration performance is also described in [4, 39, 46] and this improvement has been attributed to a variety of sources. For example, Bergen *et al.* [4] suggest that filtering improves registration performance by eliminating high frequency image content that is most likely to include the effects of aliasing. Filtering may also correct for estimation biases resulting from fixed pattern noise [11], or from biases inherent to the estimation method used [46, 47]. Elad *et al.* [22] discuss the design of filters for gradient-based motion estimation and show a way to design filters that combine smoothing and

gradient based estimation. Elad *et al.* [22] also provide a summary of the filtering technique proposed by Simoncelli *et al.* in [50] which contains a presmoothing step. The filters arrived at in these papers are essentially a combination of low-pass filters that remove extreme high-frequency content and high-pass filters that remove low-frequency image biases.

One low-pass filter design approach that has been explored in the literature is Wiener filtering two images before attempting to register them [34]. This method may also be applied to the projections of an image. Using a signal Power Spectral Density (PSD) calculated from the noise-free image (\mathcal{S}_O) and a noise PSD calculated from the known characteristics of our noise (\mathcal{S}_n), an optimal filtering kernel can be calculated for a projection as [26]

$$\mathcal{K} = \frac{\mathcal{S}_O}{\mathcal{S}_O + \mathcal{S}_n}. \quad (2.8)$$

The filtering kernel \mathcal{K} is then multiplied with the Fourier transform of the projection. The inverse Fourier transform of the result of this multiplication is the filtered projection. An exact calculation of the PSD or the autocorrelation relies on either *a priori* knowledge or estimation from noisy data using techniques such as those described by Kay [30]. Elad [22] also suggests that the spectral content of a series of images should be considered in designing a filter to improve image registration performance.

This background on filtering may be combined with a FOM that is generalized from (2.7) to design simple filters that are optimized to minimize the MSE of registration estimates. The difference between the filters proposed and Wiener (or other) filtering is that the proposed filters use binary coefficients that allow them to be implemented as integer additions in combinational logic. These simple and computationally inexpensive filters can be used to achieve effects approaching those

of more complicated Wiener filters that need to be implemented using floating-point calculations.

2.3 Bounds on the Mean-Squared Error of Estimators

The performance of an estimator used for estimating shifts between images is governed by fundamental statistical limits. A variety of approaches have been used to examine bounds on the performance of image registration algorithms; however, one of the most common approaches is the Cramer-Rao lower bound (CRLB). Robinson and Milanfar derive the CRLB on the performance of image registration algorithms in [46]. They derived related bounds in [48] for analyzing the performance of super-resolution imaging. Yetik and Nehorai provide extensive analysis and derivations of Cramer-Rao lower bounds for a variety of geometric distortion models in [55].

Van Trees shows that the CRLB on an unbiased estimate $\hat{\alpha}$ of a single non-random parameter of a vector of data \mathbf{d} is [53]

$$\text{VAR}[\hat{\alpha}(\mathbf{d})] \geq \left\{ -E \left[\frac{\partial^2 \ln p(\mathbf{d}|\alpha)}{\partial \alpha^2} \right] \right\}^{-1}. \quad (2.9)$$

Where multiple non-random parameters are estimated, a Fisher Information Matrix (FIM), \mathbf{J} can be derived. Say a vector of non-random parameters $\boldsymbol{\Theta}$ is to be estimated from a vector of observations, \mathbf{d} . Each element of \mathbf{J} located at index (i, j) can be defined [53]

$$\mathbf{J}_{i,j} \triangleq -E \left[\frac{\partial^2 \ln p(\mathbf{d}|\boldsymbol{\Theta})}{\partial \boldsymbol{\Theta}(i) \partial \boldsymbol{\Theta}(j)} \right]. \quad (2.10)$$

The phase shift between two vectors of identical data received through two different channels is an example of a parameter that is commonly estimated. For applications with low SNRs, estimates of this shift exhibit thresholding behavior [37, 41] which is not captured by the CRLB. As the SNR decreases, there may be a point in the measured MSE where shift estimation errors begin to exceed those predicted by

the CRLB even though the estimator is capable of reaching the CRLB at higher SNRs. This thresholding occurs because, as the noise in an image increases, it becomes increasingly likely that registration errors will occur at the subpeaks of the autocorrelation of an image. This behavior is observed but not quantified in Robinson and Milanfar's work on performance bounds [46].

This deficiency in quantifying this thresholding behavior can be resolved by examining another bound used traditionally in high-noise, low-signal-strength environments for estimating delays between signals. The Barankin bound has been used for time-delay estimation of one-dimensional signals [37, 38, 41]. In the literature, thresholding behavior is predicted and estimated by the Barankin bound in the one-dimensional problems of radar and sonar returns in the literature [37, 38, 41]. If the true values of a vector to be estimated are represented by the vector Θ_0 , a shifted version of the vector likely to produce an error can be represented as the vector Θ_n . Considering the most likely values of Θ_n , the Barankin bound of an unbiased estimator an estimate $\hat{\Theta}$ of a vector of true parameters Θ_0 can be written and calculated as [37]

$$\sigma^2 \geq \mathbf{J}^{-1} + (\Phi - \mathbf{J}^{-1}\mathbf{A})(\Delta^{-1})(\Phi - \mathbf{J}^{-1}\mathbf{A})^T, \quad (2.11)$$

where $\Delta = \mathbf{B} - \mathbf{A}^T \mathbf{J}^{-1} \mathbf{A}$, \mathbf{J} is the FIM calculated as in the CRLB, and where

$$\mathbf{A}_{i,j} = \mathbb{E} \left[\frac{\partial \ln p(\mathbf{d}|\Theta_0)}{\partial \Theta_0(i)} L(\mathbf{d}|\Theta_j, \Theta_0) \right], \quad (2.12)$$

$$\mathbf{B}_{i,j} = \mathbb{E} [L(\mathbf{d}|\Theta_i, \Theta_0) L(\mathbf{d}|\Theta_j, \Theta_0)], \quad (2.13)$$

$$\Phi = [\Theta_1, \Theta_2, \dots, \Theta_n], \quad (2.14)$$

and $L(\mathbf{d}|\Theta_i, \Theta_0)$ is the likelihood function

$$L(\mathbf{d}|\Theta_i, \Theta_0) = \frac{p(\mathbf{d}|\Theta_i)}{p(\mathbf{d}|\Theta_0)}. \quad (2.15)$$

The notation Θ_i is used to indicate values of Θ other than its true value Θ_0 and $\Theta_j(i)$ to indicate a scalar element of Θ_j indexed at i .

Because time-delay estimation is mathematically similar to spatial displacement estimation in images, this bound may be extended to two-dimensional spatial estimation to better predict the behavior of the effects of increasing noise on the performance of correlation-based image registration. This bound may also help to explain difficulties in estimating shifts when images are blurred as is the case with inadvertent focal-length errors.

2.4 Calculating the Optical Transfer Function of an Imaging System

Focal-length errors can be modeled using Fourier optics. In his book on this subject, Goodman describes the effects of defocus errors on an incoherent optical system [24] by describing an Optical Transfer Function (OTF) as a function of a generalized pupil function. This approach requires the use of the Amplitude Transfer Function (ATF), the wavelength (λ) of light being detected, the size and shape of the pupil, the distance between the optical system and the focal plane as input parameters (z_i), and the distance between the optical system and the plane of the detector (z_a , assuming $z_i \neq z_a$ as in [24]). Using (x, y) to indicate coordinates in the aperture plane where $(0, 0)$ is located at the center of a circular aperture of width $2l$, a pupil can be expressed as [24]

$$\mathbf{P}(x, y) = \begin{cases} 1 & : \sqrt{x^2 + y^2} \leq l \\ 0 & : \text{else} \end{cases} \quad (2.16)$$

Goodman uses this pupil function to describe the frequency-domain ATF of a system with a focal length aberration using frequency coordinates (f_x, f_y) as

$$\mathcal{G}(f_x, f_y) = \mathbf{P}(\lambda z_i f_x, \lambda z_i f_y) \exp \left[\frac{j2\pi}{\lambda} W(\lambda z_i f_x, \lambda z_i f_y) \right], \quad (2.17)$$

where

$$W(x, y) = -\frac{1}{2} \left(\frac{1}{z_a} - \frac{1}{z_i} \right) (x^2 + y^2). \quad (2.18)$$

Finally, the OTF $\mathcal{H}(f_x, f_y)$ can be calculated by first calculating the 2-D autocorrelation of the ATF and then normalizing so that the peak of the OTF equals unity. The inverse Fourier transform of \mathcal{H} will be designated \mathbf{H}_o . This optical filtering is performed before readout noise is added to the image which affects the formation of the PDF of the image in that additive noise is uncorrelated in the filtered pixels of the optically filtered image. To derive CRLBs, analysis is limited to those values of \mathcal{H} which can be reasonably approximated by non-singular \mathbf{H}_o (i.e. sub-wavelength defocus errors).

2.5 Existing Single-Frame Denoising Algorithms

In order to reduce the computational complexity or output performance of modern image denoising algorithms, it is important that one has an understanding of how these algorithms work. To this end, this section provides an overview of related image denoising algorithms that selectively average pixels within a single image. Image processing literature is rife with methods for denoising images and this review will not attempt to address all of these methods. Good overviews of traditional single-frame image denoising algorithms may be found in [9] and in [26]; however, an overview of the methods that are most similar to the algorithms proposed in this research is provided. These similar methods are those that reduce noise while attempting to maintain high spatial frequency image content.

The inadvertent reduction of high-frequency content is the Achilles' heel of many basic image processing algorithms. Some of the simplest image smoothing algorithms are linear, shift-invariant filters such as the Gaussian filter, the Wiener filter [26], and the mean-shift algorithm which replaces pixels in an image with

the mean of the pixels in a surrounding neighborhood [16, 23]. These algorithms reduce the noise power in an image through averaging; however, because they average without respect to local content, they also tend to blur high-frequency image content. This deficiency necessitates the study of more complex denoising algorithms that will maintain this high frequency content while removing additive noise from the image.

In an effort to remove noise while preserving high-frequency content, research has turned to nonlinear filters. One of the simplest and most commonly employed nonlinear methods for reducing noise power is the median filter which replaces individual pixels in an image with the median value of pixels in a defined neighborhood surrounding these pixels [26]. This filter generally provides better results than Gaussian smoothing; however, better performance may be obtained using more advanced smoothing techniques.

Neighborhood filtering is one nonlinear approach to image denoising that has shown recent promise in the literature and has produced results that improve upon those of many other image denoising algorithms. In a neighborhood filter, the algorithm assigns an output value to a pixel based on an evaluation of the relationships between pixels in a surrounding neighborhood. Many, if not most, neighborhood filters are fundamentally related and the chapter begins by highlighting these relationships as it reviews these algorithms. The algorithms reviewed include total variation minimization [49], anisotropic diffusion [40], bilateral filtering [52], the nonlocal means [9], and an optimal patch-based algorithm [31]. These algorithms all include some measure of pixel similarity that allows the algorithm to differentiate between similar and dissimilar pixels within the same image. All of the algorithms discussed do this in ways that are different but are ultimately related.

2.5.1 Total Variation Minimization. The Total Variation (TV) minimization algorithm is an iterative image smoothing algorithm that preserves edges [13, 49]. Unlike the other algorithms that will be discussed in this chapter, TV minimiza-

tion identifies similar pixels in the local neighborhood of a given pixel by examining the L1 norm of the neighborhoods surrounding these pixels. This algorithm is described for the continuous case by Chambolle [12] and by Rudin *et al.* [49] among others; however, because digital images are sampled discretely, the most relevant formulation is provided by Chan *et al.* [13]. For explanatory purposes, say there is a pixel $\mathbf{D}(x, y)$ in a larger image \mathbf{D} of size $S \times T$ where $S \in \mathbb{N}$ and $T \in \mathbb{N}$ that is indexed using $x \in \mathcal{S} \triangleq \{1, \dots, S\}$, $y \in \mathcal{T} \triangleq \{1, \dots, T\}$ and say the set of the indices of the closest neighbors of $\mathbf{D}(x, y)$ is defined as \mathcal{A} . Then, the local variation at a pixel $\mathbf{D}(x, y)$ is defined as [13]

$$|\nabla_{x,y}\mathbf{D}| = \sqrt{\sum_{(i,j) \in \mathcal{A}} (\mathbf{D}(x, y) - \mathbf{D}(i, j))^2}. \quad (2.19)$$

To avoid discontinuities in later calculations, the local variation in (2.19) is modified to be the regularized local variation

$$|\nabla_{x,y}\mathbf{D}|_a = \sqrt{|\nabla_{x,y}\mathbf{D}|^2 + a^2}, \quad (2.20)$$

where a is small constant normally chosen on the order of 10^{-4} . The output of the filtering is then

$$\hat{\mathbf{I}}(x, y) = \sum_{(i,j) \in \mathcal{A}} h_{\alpha\beta}(\mathbf{D}(i, j))\mathbf{D}(i, j) + h_{\alpha\alpha}(\mathbf{D}(i, j))\mathbf{D}(x, y) \quad (2.21)$$

where between two pixels α indexed at $\mathbf{D}(i, j)$ and β at $\mathbf{D}(x, y)$

$$\begin{aligned} h_{\alpha\beta}(\mathbf{D}(i, j)) &= \frac{w_{\alpha\beta}(\mathbf{D}(i, j))}{\lambda + \sum_{(x,y) \in \mathcal{A}} w_{\alpha\beta}(\mathbf{D}(x, y))}, \\ h_{\alpha\alpha}(\mathbf{D}(i, j)) &= \frac{\lambda}{\lambda + \sum_{(i,j) \in \mathcal{A}} w_{\alpha\beta}(\mathbf{D}(x, y))}, \\ w_{\alpha\beta}(\mathbf{D}(i, j)) &= \frac{1}{|\nabla_{x,y}\mathbf{D}|_a} + \frac{1}{|\nabla_{i,j}\mathbf{D}|_a}, \end{aligned}$$

and where λ is a Lagrange multiplier chosen for a known additive noise of variance σ^2 as

$$\lambda = \frac{1}{\sigma^2} \frac{1}{ST} \sum_{x \in \mathcal{S}, y \in \mathcal{T}} \sum_{(i,j) \in \mathcal{A}} w_{\alpha\beta} (\mathbf{D}(i,j) - \mathbf{D}(x,y)) (\mathbf{D}(x,y) - \mathbf{D}(x,y)^0). \quad (2.22)$$

Since this is an iterative algorithm, the notation \mathbf{D}^0 is used to indicate the original value of the image. It is interesting to note, as do Chan *et al.* [13], that this algorithm gives large weights to those pixels in areas with low variation and large weights to those with low variation. The effect of this is that area smoothing occurs with less degradation of true image edges than would be found with a linear filter. This basic mechanism is also present in anisotropic diffusion which is examined in the next section.

2.5.2 Anisotropic Diffusion. Another method of image smoothing that attempts to minimize degradation of image edges is anisotropic diffusion [40]. This method attempts to iteratively smooth an image by treating the smoothed image as solutions to the heat equation over arbitrary time steps. Perona and Malik [40] create an anisotropic diffusion algorithm that accounts for edges and preserves edges in the smoothed image by assuming them to be differences in conductivity for heat flow. For an iteration of the algorithm on a single pixel in an image \mathbf{D} ,

$$\hat{\mathbf{I}}(x, y, t) = \text{div}(c(x, y, t) \nabla \hat{\mathbf{I}}(x, y, t - 1)), \quad (2.23)$$

where $c(x, y, t)$ indicates the conductivity between pixels, t is an arbitrary time step, $\hat{\mathbf{I}}(x, y, 0) = \mathbf{D}(x, y)$, div indicates the divergence of a gradient, and ∇ is a function that produces the image gradient. The algorithm preserves edges in the image by modeling conductivity between pixels as functions of the image gradient so that the conductivity between a pixel at a point (x, y) and its neighbors at time t is

$c(x, y, t) = g(\nabla \mathbf{I})$ where $g(\nabla \mathbf{I})$ is proposed variously by Perona and Malik [40] as

$$\begin{aligned} g(\nabla \mathbf{I}) &= \exp\left(\frac{-\|\nabla \hat{\mathbf{I}}\|}{\sigma^2}\right), \\ g(\nabla \mathbf{I}) &= \frac{1}{1 + \left(\frac{\|\nabla \hat{\mathbf{I}}\|}{\sigma}\right)^2}, \end{aligned}$$

and by Black *et al.* [6] as

$$g(\nabla \mathbf{I}) = \begin{cases} \frac{1}{2}(1 - \|\nabla \hat{\mathbf{I}}\|^2)^2 & : |\nabla \hat{\mathbf{I}}| \leq \sigma \\ 0 & : else \end{cases}. \quad (2.24)$$

where σ is a scaling constant in all of the above equations. As was the case with TV smoothing, the effect of this algorithm is to facilitate smoothing between pixels in regions with similar intensities, and to inhibit smoothing between pixels in dissimilar regions. In this case, the algorithm assigns low conductivity values to pixel values across large gradients, and high conductivity values to pixels across small image gradients. This again helps to preserve the edges in the image by determining similar and dissimilar pixels adjacent to a single pixel under consideration. The next algorithm attempts to perform nonlinear smoothing using a non-iterative method.

2.5.3 Bilateral Filtering. Non-iterative algorithms are generally preferable to iterative algorithms because they can be performed with improved processing time. Bilateral filtering [21,52] is a non-iterative smoothing method that works by applying a filtering kernel to an image that is designed to avoid smoothing across edges within the image. It does this using a filtering kernel that averages pixels according to two measures of distance: spatial distance and radiometric distance. The part of the kernel that examines spatial distance works like a traditional Gaussian filter and includes pixels in the average that are spatially close to a given pixel. The part of the kernel that examines radiometric distance includes pixels in the average that are similar in intensity. As a discrete example, say that there exists an image \mathbf{D} of size

$S \times T$ where $S \in \mathbb{N}$ and $T \in \mathbb{N}$ that is indexed using $i \in \mathcal{S} \triangleq \{1, \dots, S\}$, $j \in \mathcal{T} \triangleq \{1, \dots, T\}$. Then, it is possible to find a smoothed estimate of a pixel as

$$\hat{\mathbf{I}}(i, j) = \frac{1}{Z(i, j)} \sum_{u \in \mathcal{S}} \sum_{v \in \mathcal{T}} \mathbf{D}(u, v) f_s((i, j), (u, v)) f_r(\mathbf{D}(i, j), \mathbf{D}(u, v)). \quad (2.25)$$

In this equation, $f_s((i, j), (u, v))$ is a function that provides a measurement of the spatial distance between pixels, $f_r(\mathbf{D}(i, j), \mathbf{D}(u, v))$ is a function that measures the radiometric distance between pixels, and $Z(i, j)$ is a normalizing factor. A Gaussian example provided by Tomasi and Manduchi [52] gives these as

$$\begin{aligned} f_s((i, j), (u, v)) &= \exp \left(-\frac{1}{2\sigma_s^2} ((i - u)^2 + (j - v)^2) \right), \\ f_r(\mathbf{D}(i, j), \mathbf{D}(u, v)) &= \exp \left(-\frac{1}{2\sigma_r^2} ((\mathbf{D}(i, j) - \mathbf{D}(u, v))^2) \right), \\ Z(i, j) &= \sum_{u \in \mathcal{S}} \sum_{v \in \mathcal{T}} f_s((i, j), (u, v)) f_r(\mathbf{D}(i, j), \mathbf{D}(u, v)), \end{aligned} \quad (2.26)$$

where σ_s and σ_d are empirically-determined constants that spread the kernels to achieve the desired level of filtering.

The effect of combining these distance measures is that pixels that are spatially close but not close in intensity (as on an edge) are not included in the average. Interestingly, Elad [21] showed that the bilateral filter is actually mathematically equivalent to anisotropic diffusion, a result that is suggested by the work of Barash [2]. This result provides a mathematical connection to the next smoothing algorithm described in this chapter.

2.5.4 Nonlocal Means. Another recent approach to image denoising is the nonlocal means (NLM) algorithm proposed by Buades *et al.* [8, 9]. This algorithm compares and averages pixels chosen by evaluating their similarity based on the similarity of their local neighborhoods. Based on this measure of similarity, this algorithm assigns weights to other pixels of the same image and then uses a weighted

sum approach to arrive at a denoised version of the pixel of interest. Kervrann and Boulanger [32] provide the link to some of the other algorithms in this chapter when they note that when the neighborhood size used in the NLM to compare pixels is equal to 1, the NLM algorithm reduces to the bilateral filter. The unique aspect of this filter, however, is that it measures similarity between pixels by comparing the neighborhoods around the pixels, rather than examining the pixel values themselves.

By way of explanation, suppose there again exists an image \mathbf{D} of size $S \times T$ where $S \in \mathbb{N}$ and $T \in \mathbb{N}$. For a given pixel located at (i, j) , the neighborhood around the pixel can be represented as an $N \times N$ -sized subimage $\mathbf{F}_{i,j} \subset \mathbf{D}$ where $\mathbf{F}_{i,j}$ is centered at $i \in \mathcal{S} \triangleq \{1, \dots, S\}$, $j \in \mathcal{T} \triangleq \{1, \dots, T\}$ and where $\{N = 2n + 1 \mid n \in \mathbb{N}\}$. If a zero pad is added to image \mathbf{D} by n in all directions then, for all $s \in \mathcal{S}$ and $t \in \mathcal{T}$, there are $N^2 - 1$ other subimages in \mathbf{D} which may be similar in an L2-norm sense to F . These subimages are denoted as $\mathbf{G}_{s,t}$. The NLM algorithm then assigns weights $\mathbf{A}(s, t) \in \mathbb{R}$ to each $\mathbf{G}_{s,t}$ and constructs a denoised version of the center pixel of $\mathbf{F}_{i,j}$ as

$$\text{NLM}(\mathbf{D}(i, j)) = \sum_{s \in \mathcal{S}, t \in \mathcal{T}} \mathbf{A}(s, t) \mathbf{G}_{s,t}(n + 1, n + 1), \quad (2.27)$$

where the center pixel of $\mathbf{G}_{s,t}$ is found at the block coordinates $(n + 1, n + 1)$. For the case where $s \neq i$ and $t \neq j$, the weights $\mathbf{A}(s, t)$ are assigned by the following exponential function

$$\mathbf{A}(s, t) = \frac{1}{\mathbf{z}(i, j)} \exp \left(-\frac{\|\mathbf{F}_{i,j} - \mathbf{G}_{s,t}\|_F^2}{h^2} \right), \quad (2.28)$$

where

$$\mathbf{z}(i, j) = \sum_{s \in \mathcal{S}, t \in \mathcal{T}} \exp \left(-\frac{1}{h^2} \|\mathbf{F}_{i,j} - \mathbf{G}_{s,t}\|_F^2 \right). \quad (2.29)$$

In this equation, as in anisotropic diffusion, h is an experimentally determined parameter that controls the roll off of the exponential function. For the case where $s = i$ and $t = j$, the weight $\mathbf{A}(s, t) = \max\{\mathbf{A}(s, t) | s \neq i, t \neq j\}$ is used.

Other than the aforementioned difference in similarity measurement, the NLM algorithm is also interesting because it appears in [9] to provide performance that is superior to the algorithms discussed thus far. However, it also has several facets which make it suboptimal:

- No analytic method is presented for determining the parameter h used to control the decay of the exponential.
- Features in natural images are frequently similar but may be different in illumination or reflectivity. The NLM algorithm's reliance on the L2 norm of the error between \mathbf{F} and \mathbf{G} does not exploit these highly similar regions which could be separated only by constant bias or gain.
- The algorithm produces results that contain false contouring similar to that seen when image pixels are sub-optimally quantized.

These shortcomings are addressed and ameliorated in the next algorithm.

2.5.5 Patch-Based Denoising with Optimal Spatial Adaptation. Kervrann and Boulanger [31, 32] describe the Optimal Spatial Adaptation (OSA) algorithm that overcomes some of the shortcomings of the NLM algorithm. They describe a block-based algorithm that thresholds the Euclidian distance between blocks and builds an exponentially weighted average of similar blocks. Their algorithm eliminates the arbitrary parameter in Buade's algorithm by employing an adaptively-sized window around a given pixel to find an optimal neighborhood size. Like the NLM algorithm, their method uses an exponential weighting of candidate blocks; however, it improves on Buade's method by eliminating the arbitrary parameter from their algorithm and reducing smoothing artifacts. Kervrann and Boulanger's papers also

describe a method for precisely estimating the error function of similar regions based on the chi-square distribution.

2.5.6 Other Patch-Based Methods. Other methods have been proposed that employ matching and combine blocks using different measures of similarity. Dabov *et al.* describe a method for block matching and filtering in the Fourier domain [17]. Their method selects similar blocks by hard-thresholding the 2-D transforms of two blocks and then examining the L2 distance between the thresholded transformed blocks. They then create a three-dimensional array of the matching blocks, and perform a 3-D transform, thresholding, and inverse transform operation to arrive at denoised estimates of the matched blocks.

The Unsupervised Information-Theoretic Adaptive Filter (UINTA) proposed by Awate and Whitaker [1] also combines information from blocks within an image in a way that is unique. Their algorithm selects a random sample of blocks in an image and, using the L2 distance between the sampled blocks and a block of interest, creates an entropy estimate for the block of interest. It then uses an iterative gradient-descent algorithm to reduce the entropy of the center pixel in the block of interest. The algorithm iterates until the variance of the residual error between the denoised block and the observed block equals the known variance of the noise. Although this algorithm does produce good results, it is computationally intensive and has a tendency to over smooth image content. It is noted to work best on images that are highly periodic in content [9].

2.6 Chapter Summary

This chapter has provided a review of current literature on image registration and denoising that yields a number of opportunities for research. The review discussed potential improvements to existing registration methods that may be used to improve image registration performance. It also showed an opportunity to employ

the Barankin bound to extend current understanding of image registration algorithm performance. It introduced methods employing Fourier optics that will be used to model focal-length errors in optical systems. Lastly, it provided a review of related nonlinear image filters that focused on the measures of similarity they use to identify similar pixels and regions within images.

III. Improving Projection-Based Image Registration

This chapter describes a method for accomplishing fast, reliable image registration that may be implemented with a low computational cost. The method described extends work on projection-based image registration performed using cross correlations described first by Cain *et al.* [11] and reviewed in Section 2.1. The goal of this chapter is to provide a method for designing a low-pass filter that minimizes the mean-squared error of registration estimates.

The filters are designed to exploit local correlations within images. Most naturally-occurring images captured by imaging systems exhibit some local spatial correlation [11] and a variety of models can be used to approximate correlation in images [26]. This spatial correlation, however, is not guaranteed and is not necessarily locally consistent within an image. In particular, spatial correlation may be absent in natural images of free space and in images that are under sampled. This chapter demonstrates that if significant spatial correlation is present in the image projections, the performance of projection-based image registration algorithms can be improved significantly by the application of simple low-pass filters.

The chapter assumes that the images to be registered are wide-sense stationary (WSS) and ergodic which allows the removal of spatial indices from second order statistics in the evaluation of the expected value of an image using a spatial average. In practice, many images have unevenly distributed intensities that do not meet the WSS requirement for a constant mean. Section 4.3 discusses bias reduction techniques which allow the use of a WSS assumption even when the intensities are not evenly distributed.

If speed is a concern, a simple bias-reduction filter can be implemented optically using a two-lens system. If one lens is defocused so it acts as a low-pass filter, the difference between the two images (or two image projections) is the high-frequency content of the image. This arrangement is simulated in experiments

with aerial imagery where it is apparent that this is essentially an optical edge detection system. This chapter does not attempt a detailed description of the design of the the bias reduction portion of such a filtering arrangement except to note that an optical filter combined with the type of filter introduced here is much less computationally complex than any of the methods described in the literature thus far.

It is demonstrated in this chapter that these simple and computationally inexpensive filters can be used to achieve effects approaching those of more complicated Wiener filters that need to be implemented using floating-point calculations.

3.1 Improved Projection-Based Algorithm

When the environment is expected to yield images with significant local spatial correlation, and an imaging system is designed to provide images that are sampled sufficiently to detect this correlation, these facts can be exploited to design filters that can improve the performance of a projection-based shift estimator.

The low-pass filters designed in this chapter are convolutional kernels that are applied to the projections of two images prior to calculating their cross-correlation. Assuming that the spatial correlation of an image is the same in both projections, a filtering kernel of length w can be determined for both dimensions as

$$\mathbf{h}_w(z) = \sum_{n=0}^{w-1} \delta(z - n). \quad (3.1)$$

For each projection $\mathbf{d}_{n,x}$ and $\mathbf{d}_{n,y}$, the filtered projections are then calculated

$$\mathbf{f}_{n,x} = \mathbf{d}_{n,x} * \mathbf{h}_w, \quad (3.2)$$

$$\mathbf{f}_{n,y} = \mathbf{d}_{n,y} * \mathbf{h}_w, \quad (3.3)$$

For imagery exhibiting spatial correlation, the effect of the aforementioned filtering is to increase the effective SNR of the individual projections. In order to mitigate the effects of new information entering the scene, one of the projections is windowed using the method described in [11]. Using the notation $\bar{\mathbf{f}}_{i,y}$ to indicate a vector with all elements equal to the mean of the filtered data within the windowed region, these modified projections are then used to compute the 1-D cross correlations of the projections of two images as

$$\mathbf{p}_y = (\mathbf{f}_{1,y} - \bar{\mathbf{f}}_{1,y}) * \mathbf{W}_f(\mathbf{f}_{2,y} - \overline{\mathbf{W}_f \mathbf{f}_{2,y}}), \quad (3.4)$$

$$\mathbf{p}_x = (\mathbf{f}_{1,x} - \bar{\mathbf{f}}_{1,x}) * \mathbf{W}_f(\mathbf{f}_{2,x} - \overline{\mathbf{W}_f \mathbf{f}_{2,x}}). \quad (3.5)$$

In terms of the original data, these cross correlations can be written as

$$\mathbf{p}_y = (\mathbf{d}_{1,y} * \mathbf{h}_w - w\bar{\mathbf{d}}_{1,y}) * \mathbf{W}_f(\mathbf{d}_{2,y} * \mathbf{h}_w - w\bar{\mathbf{d}}_{2,y}), \quad (3.6)$$

$$\mathbf{p}_x = (\mathbf{d}_{1,x} * \mathbf{h}_w - w\bar{\mathbf{d}}_{1,x}) * \mathbf{W}_f(\mathbf{d}_{2,x} * \mathbf{h}_w - w\bar{\mathbf{d}}_{2,x}). \quad (3.7)$$

As in the unfiltered case, the shift estimates are computed from these projections as

$$\hat{\alpha} = \arg \max_z |\mathbf{p}_y(z)|, \quad (3.8)$$

$$\hat{\beta} = \arg \max_z |\mathbf{p}_x(z)|. \quad (3.9)$$

The length of the filter w is chosen by using a FOM. In the next section the FOM introduced in [11] is modified so that it can be used as a tool to design a low pass filter that minimizes the MSE of registration errors.

3.1.1 Introduction of the Revised Figure of Merit. Although the FOM introduced in [11] and reviewed in Section 2.1 was shown to be effective in evaluating the performance of both 2-D cross correlation and projection-based registration algorithms, it is necessary to modify it for use with the new algorithm. The problem

lies in measuring the variance of the terms in the denominators of (2.6) and (2.7). In the new algorithm, applying a convolutional kernel of any size other than one to the projections leads to a 1-D cross correlation function with spatially correlated noise. Since the noise is no longer independent and identically distributed (i.i.d.) in the points on the cross-correlation, the denominator in Cains FOM no longer accurately reflects the vertical distance between two points on the cross correlation. Therefore the modified figures of merit are used where again, the notation $E[.]$ indicates the expected value of a random variable and z is used as an integer index for the cross correlation:

$$F_{P_y}(z, \alpha) \triangleq \begin{cases} \frac{(E[\mathbf{p}_y(\alpha)] - E[\mathbf{p}_y(z)])^2}{E[\text{VAR}[\mathbf{p}_y(z) - \mathbf{p}_y(\alpha)|\mathbf{i}]]} & : z \neq \alpha \\ 0 & : \text{else} \end{cases}, \quad (3.10)$$

$$F_{P_x}(z, \beta) \triangleq \begin{cases} \frac{(E[\mathbf{p}_y(\beta)] - E[\mathbf{p}_y(z)])^2}{E[\text{VAR}[\mathbf{p}_y(z) - \mathbf{p}_y(\beta)|\mathbf{i}]]} & : z \neq \beta \\ 0 & : \text{else} \end{cases}. \quad (3.11)$$

It is important to recognize that in changing the FOM, a discontinuity has been introduced at the point $z = \alpha$ which is accounted for by assigning a value of zero at the point of discontinuity.

3.1.2 Use of the FOM in Filter Design. The correlation-based image registration involves searching an area of interest within a frame for content that is identical to a previous frame. In the region that this content is identical, the cross correlation approximates the autocorrelation of the region of interest. Thus, for any given projection, the point of primary interest is the apex of the autocorrelation which occurs at $\alpha = 0$ and indicates the most likely estimate of the point where the two frames are aligned. Using the notation $\langle \cdot, \cdot \rangle$ to denote the inner product of two vectors, if the true shift between the images is zero, then for any shift value α an error occurs when $\langle \mathbf{d}_{1,y}, \mathbf{d}_{2,y} \rangle < \langle \mathbf{d}_{1,y}, \mathbf{T}_\alpha \mathbf{d}_{2,y} \rangle$. For most common correlation models, for $\alpha \neq 0$ the L2 distance between $\mathbf{d}_{1,y}$ and $\mathbf{T}_\alpha \mathbf{d}_{2,y}$ is be smallest in expectation at

$\alpha = -1$ and $\alpha = +1$. Therefore, for an unbiased estimator, and a symmetric autocorrelation model, autocorrelation points at $\alpha = +1$ and $\alpha = -1$ are the points that are most likely to cause a registration error (i.e. these are normally the points closest in magnitude to the peak of the autocorrelation.) This approach is similar to the process of finding test points for calculation of the Barankin bound on a registration estimate [37] and furthermore, the choice of $\alpha = +1$ and $\alpha = -1$ does imply some amount of *a priori* knowledge of the structure of the signal PSD. However, this correlation structure is common to most natural images except those with flat or impulse-shaped autocorrelations.

The consequence of this conclusion is that the design of a filter can be approached by attempting to minimize the most probable registration errors. This is done by deriving a FOM for the filtered projections that measures the effective SNR between shifts of $\alpha = 0$ and $\alpha = -1$. Using the assumption that noise is i.i.d. in each pixel, it can be shown that if the projections are filtered with a kernel of size w and a windowing function of size L , the FOM for the filtered projections of an $N \times N$ image is

$$F_{P_z}(0, 1) \triangleq \frac{(LN^2 (\text{VAR}(\mathbf{I}) - \overline{\text{COV}}_z(\mathbf{I})))^2}{w^2 \sigma^2 LN^2 (2(\text{VAR}(\mathbf{I}) - \overline{\text{COV}}_z(\mathbf{I})) + \sigma^2)}. \quad (3.12)$$

The derivation for 3.12 is included as an appendix in Sec. A.1. With an analytic expression for F_{P_y} given by (3.12), an exhaustive search of values can be performed over a reasonable range (e.g. $[1, 20]$) to find a value for w that maximizes the FOM. By maximizing the FOM with this value, the most probable registration errors can be minimized, thereby minimizing the MSE of the registration errors overall. The optimal filter size can then be written as

$$w_{y,opt} = \arg \max_w F_{P_y}(0, -1). \quad (3.13)$$

3.2 Observing Covariance in Images

In order to employ (3.12), it is necessary to have a mechanism for estimating the average covariance, $\overline{\text{COV}}_z(\mathbf{I}|\alpha)$ described in Appendix B. This section describes how this value can be estimated in an image corrupted by noise. The effects of biases on these measurements are also discussed along with suggested methods for minimizing the effects of these biases. The covariance measured and graphed in Figure 3.2 was



Figure 3.1: 1024×1024 grayscale image of the Pentagon from <http://sipi.usc.edu/database/>.

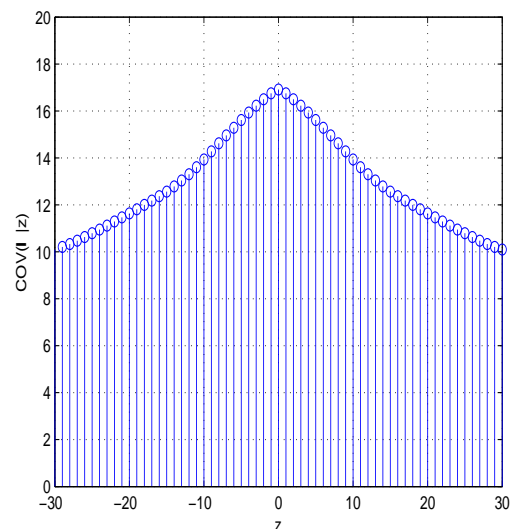


Figure 3.2: Measured covariance of the column projections of the Pentagon image in Figure 3.1.

calculated by taking the inner product of a projection and a circularly shifted version of the projection for the image shown in Figure 3.1 in the absence of noise. If the covariance model is known *a priori*, this data in can be used in calculations; however, for many applications, this information is not available. When noise is uncorrelated, the available data can be used to estimate a working covariance model using only very basic calculations.

When uncorrelated noise is added to the image, the magnitude of the center of the covariance plot (corresponding to the variance of the noise) increases. However, the effect on the off-center values of the covariance function is much less pronounced.

This effect is shown pictorially in Figs. 3.3 and 3.4 where the covariance models have been calculated by taking the inner product of a projection within the windowed area with a circularly shifted version of itself. Although a variety of techniques could be employed to estimate the optimal covariance model from the noisy covariance model, a crude but usually effective method is shown here. The vector of values representing the covariance model is called \mathbf{C} , and the index of the midpoint of \mathbf{C} is 0 then replace $\mathbf{C}(0)$ with

$$\mathbf{C}(0) = \mathbf{C}(-1) + \frac{|\mathbf{C}(-1) - \mathbf{C}(-2)|}{2}. \quad (3.14)$$

The factor of one half in (3.14) gives a very conservative estimate for the location of the peak which becomes important in the design a filter that minimizes the MSE of the registration estimate. At this point, a variety of filtering techniques could be employed to further smooth the covariance estimate. For example, when the observed scene is expected to change little between frames, a covariance model could be developed as an average of covariance models over many frames of data. However, for simplicity, results are shown using a covariance model derived from a single frame of data. With this covariance model, a filter can be designed for a given image that minimizes registration errors using (3.13).

3.3 *Experimental Results*

The algorithm was tested using a variety of standard test images and also with a series of images from an aerial imaging platform. Using standard images, the operation of the algorithm was verified with images that were heavily corrupted with AWGN. Using a sequence of frames with more typical noise values, an examination of how the algorithm would work under more common conditions such as those frequently encountered with infrared imaging systems was performed.



Figure 3.3: 512×512 Brodatz grass image from <http://sipi.usc.edu/database/>.

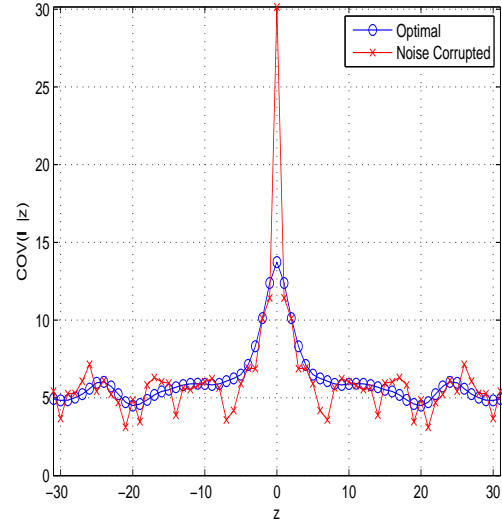


Figure 3.4: Graph of measured covariance of column projections of the grass image in Figure 3.3 calculated without noise and again with AWGN of $\sigma = 100$ (PSNR = 8.12).

In a first set of experiments, the algorithm was tested using a variety of images that were severely corrupted by adding AWGN to two frames with $\sigma = 100$ (PSNR = 8.14). Although this type of SNR is not normally encountered in high-illumination imaging, it is not unusual for LADAR, or passive infrared systems under examination for military applications. For test purposes, this level of corruption also produces errors in sufficient quantities for meaningful analysis.

Using these corrupted images, the improvement in registration accuracy attained by the registration algorithm was measured by estimating shifts for two frames when the actual shift was zero. The correlation function for the images was measured by circularly shifting windowed, noise-free projections of the images and also by using a single noisy frame of data and the adjustment in (3.14). In an actual imaging system, an expected correlation model may also be assumed based on *a priori* knowledge of the given imaging system and likely observations.



Figure 3.5: 1024×1024 Brodatz sand image from <http://sipi.usc.edu/database/>.

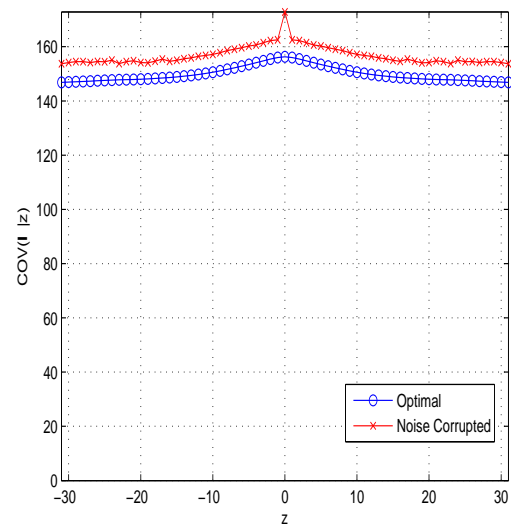


Figure 3.6: Graph of measured covariance of column projections of the sand image in Figure 3.5 calculated without noise and again with AWGN of $\sigma = 100$ (PSNR = 8.13). Although an offset is evident between the two cases, this does not affect the results of the calculations.

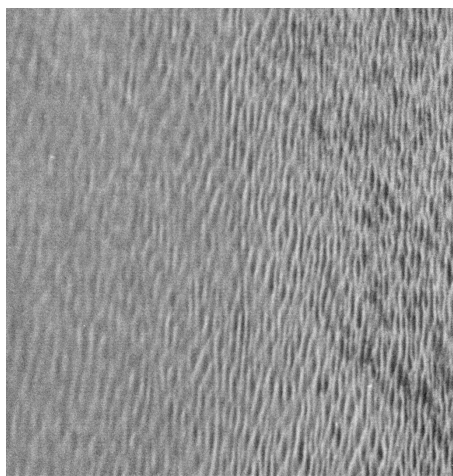


Figure 3.7: 1024×1024 Brodatz water image from <http://sipi.usc.edu/database/>.

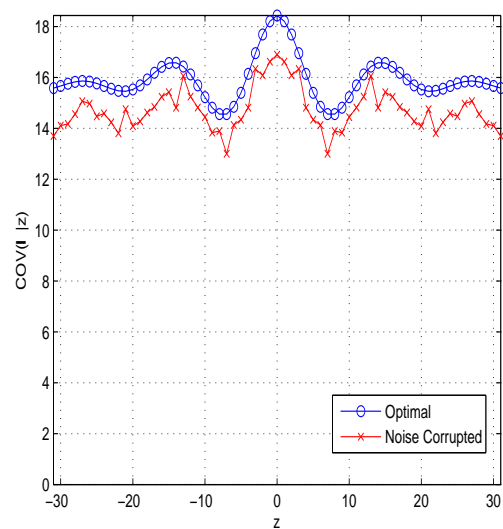


Figure 3.8: Graph of measured covariance of column projections of the image in Figure 3.7 calculated without noise and again with AWGN of $\sigma = 100$ (PSNR = 8.13). In this graph the peak has been adjusted using (3.14).

Image	MSE unfiltered	MSE $\hat{F}_{P_y, max}$	MSE Wiener	MSE $\hat{F}_{P_y, max}$ noisy	MSE _{min}	w $\hat{F}_{P_y, max}$	w $\hat{F}_{P_y, max}$ noisy	w MSE _{min}
Pentagon (Figure 4.1)	1.42	0.384	0.381	0.384	0.340	11	11	9
Grass (Figure 3.3)	0.284	0.112	0.101	0.127	0.112	3	2	3
Water (Figure 3.7)	0.557	0.084	0.064	0.102	0.084	5	4	5
Sand (Figure 3.5)	0.545	0.135	0.119	0.195	0.080	7	10	5
Tank (Figure 3.15)	6.459	3.90	1.945	3.90	1.850	2	2	12

Table 3.1: Measured and predicted results of the improved registration algorithm.

In the following discussion and graphs, the degree of noise in an image is measured using Peak Signal-to-Noise Ratio (PSNR) where the PSNR of an $S \times T$ sized image is defined as:

$$PSNR = 10 \log_{10} \left(\frac{255^2}{\frac{1}{ST} \|(\mathbf{I} - \hat{\mathbf{I}})\|_F^2} \right) \quad (3.15)$$

where \mathbf{I} is the diffraction-limited image, $\hat{\mathbf{I}}$ is the corrupted version of the image, and $\|\dots\|_F$ is the Frobenius norm which is defined as the square root of the sum of the squares of the elements of a matrix. The test images studied were 256-level gray scale so 255^2 was used uniformly as the maximum pixel value in the numerator.

For each image the algorithm evaluated kernel sizes from 1 to 30. 1000 pairs of frames were produced (i.e. a total of 30,000 trials) for each image and the shift was estimated from the projections of the two frames. As expected, the results were dependent upon the covariance functions of the images studied.

For comparison purposes, results were also obtained by using the optimal low-pass filter of the projections as computed by Wiener filtering. For the images studied, the inverse Fourier transforms of these images were approximately triangular or sinc-like. Representative filtering kernels for column projections of the Pentagon and Brodatz water images are shown in Figure 3.13 and Figure 3.14. MSEs resulting from filtering using these kernels are found in Table 4.1.

The results are summarized in Table 4.1. As shown in this table, an improvement in the MSE of the registration estimate was demonstrated for all of the images

considered. The results also showed that for most images, the shape of covariance function could be estimated to useable accuracy using only the covariance measured from the noisy data and (3.14). Using an average covariance function developed from multiple frames of data, it is expected that results could approach the optimal predicted results.

In the case of the Pentagon image of Figure 4.1, using a noise-free estimate of the covariance function, a filter size of 11 was predicted to produce registration estimates with minimum MSE. In fact, this value improves the MSE of the registration estimate by a factor of 3.7. Using measured data, the optimal filter size is indicated to be 9 and improves the MSE of the registration estimate by a factor of 4.18.

Results for the textured images like the grass, sand and water images in Figs. 3.3, 3.5, and 3.7 were the most accurate of the images examined as these images were the closest to being wide-sense stationary without modification. In the grass and water images, minimum MSEs occurred at kernel sizes that were predicted using noise-free estimates of the covariance functions. In the sand image the noise-free estimate of the optimal kernel size was 7 and the actual optimal kernel size was 5. Using noisy estimates of the covariance functions for these images, it was possible to improve the MSEs of the registration estimates by factors of 2.5 to 5.5 over estimates using unfiltered projections. Estimates developed using noise-free data improved the MSEs of the registration estimates by factors of 2.5 to 6.6.

Two main sources of error were evident in the experiments. A primary source of error was local image bias. A model of spatial correlation developed using circular shifting of a windowed image incurs bias at shifts other than zero as new information enters the window. In the experiments, bias was manifested as a difference between the covariance function obtained using circular shifting and the covariance function obtained by actually shifting the windowed portion of the image into regions outside the image. This bias helps to explain differences between the minimum MSE predicted using the covariance function and the measured minimum MSE. Bias was

most evident in those images, like the Pentagon image, that had significant localized feature content. Images without significant bias tended to be images of regular patterns such as the textured images in Figs. 3.3, 3.5, and 3.7.

A second source of error between kernel size and lowest MSE was the way the estimated shift was calculated. The algorithm is designed to minimize errors occurring at a of -1 or 1 which was the shift most likely to produce an error. However, statistically, errors produced by other shifts that contribute to MSE can be expected. As the covariance function of an image becomes flatter, errors contributed by shifts other than -1 and 1 makes up a larger proportion of the total errors contributing to the MSE. A more complete model could account for these additional shifts.

Estimates of F_{P_y} were, in general, less reliable for small kernel sizes (i.e. less than approximately four.) This error was attributable to errors in estimating the peak value of the covariance function and to differences between actual data values and the averaged data values used in the covariance functions.

A final significant observation involved images like the one shown in Figure 3.15. The covariance function for this image is shown in Figure 3.16. The covariance function for this image is much flatter than the others shown in this section. Although there is a significant amount of covariance present in the image, the slight slope of the covariance leads to a small value in the numerator of (3.12). As demonstrated by the graph of Figure 3.17, this image has low values for F_{P_y} , compared with other images in this section and filtering does not appreciably improve the MSE of projection-based image registration for this image nor does it change the FOM to the degree evident in the other images. This suggests that algorithm performance is dependent not only on the magnitude of the covariance function but is perhaps also dependent on the second derivative of the covariance function. This relationship will be explored in future research.

In a second set of experiments, the operation of the algorithm with a series of images taken by an actual aerial imaging system was examined. In these images,

a two-lens bias reduction system was simulated. Before registration, the images were preprocessed to detect and remove significant local biases caused by uneven illumination and glare; however, high-frequency content was left intact. For the types of illumination differences in the images, this was achieved by using very coarse low-pass filtering to identify local image intensities, subtracting the result from the original image, and then adjusting the image mean to the mean pixel value for the ensemble of images. To this result AWGN was added with $\sigma = 20$ and the ability to predict kernel size between similar frames was examined. Many frames included significant homogeneous texture with some minor features such as shadows, roads, drainage ditches and fences.

One frame was selected and a Monte Carlo simulation was performed that measured the MSE for kernel sizes from 1 to 15 for 100 trials each (1500 total trials). For this frame the calculated optimal kernel size was 4 which yielded an MSE of zero. This result is shown graphically in Figure 3.19. Using data that included the AWGN, verification was performed to confirm that FOM calculations predicted an optimal kernel size of four. Other images in the series were examined to determine whether FOM calculations performed using one frame could be used in other frames in the series.

A frame from the series with similar but different image content is Figure 3.20. Calculations of kernel size for Figs. 3.18 and 3.20 are shown in Figure 3.21. In a series of 56 images, that the mean calculated kernel size was 3.6 with a standard deviation of 1.89 and, as expected, those images that had the most similar content were the most likely to have F_{P_y} , that were closely matched in shape if not in magnitude.

3.4 Chapter Summary

This chapter has provided a method for improving the performance of projection-based image registration algorithms at minimal computational cost. It also explains how low-pass filtering can exploit spatial correlations to improve the performance of

image registration algorithms. The preceding sections have described an algorithm that improves the performance of current projection-based image registration and have described methods for choosing optimal parameters for the algorithm based on measured data from the images being registered. They have also described experiments conducted with actual test data that have confirmed the analytical results.

Transform-domain operations provide one mechanism for registering images that are not only translated but also scaled or rotated; however, changes in scale or rotation may also be detected and accounted for spatially. Use of the filtering method described in this chapter with dilated or rotated images is one possible extension to this research.

The correlation theory contained in this chapter may also be applied to a host of other applications. An obvious extension to the work contained in this paper is studying the effect that filtering has on two-dimensional correlation problems.

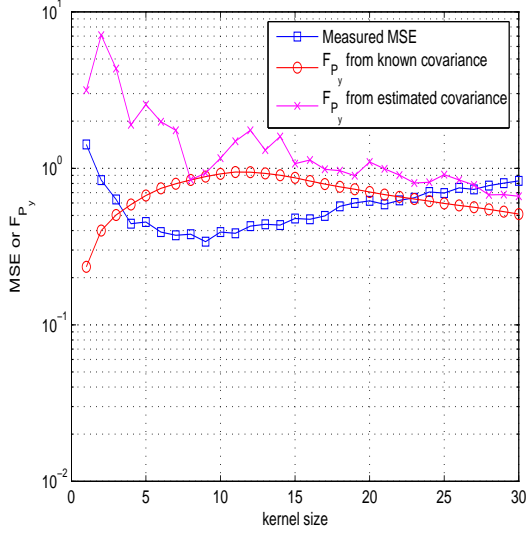


Figure 3.9: Calculated F_{P_y} , and MSE for the Pentagon image in Figure 4.1 with $\sigma_{noise} = 100$, actual shift = 0. Note the skewing of the estimates of F_{P_y} , caused by estimation errors at the peak covariance value. Errors are less evident with additional averaging at larger kernel sizes.

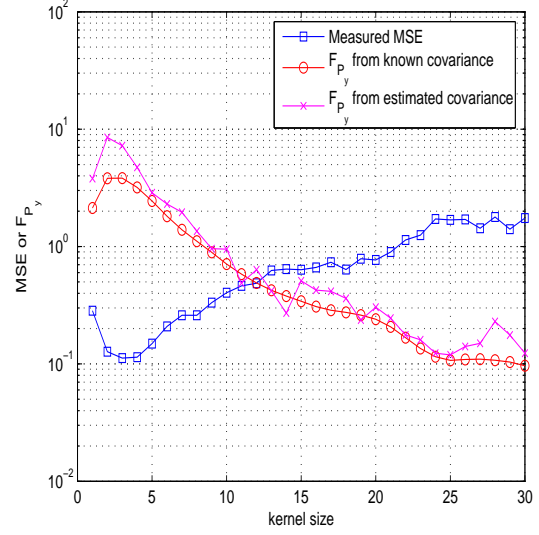


Figure 3.10: Calculated F_{P_y} , and MSE for the grass image in Figure 3.3 calculated without noise and again with AWGN of $\sigma = 100$.

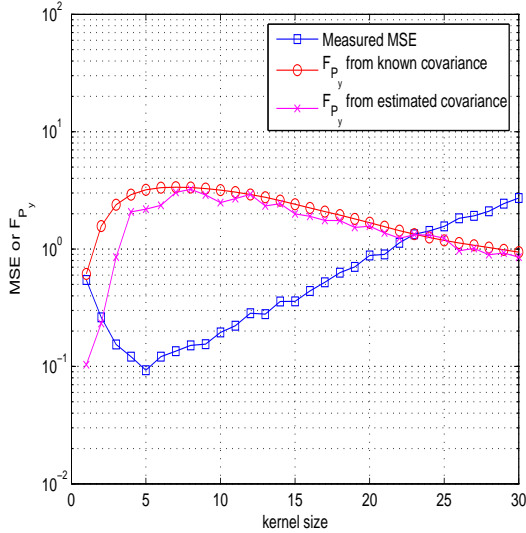


Figure 3.11: Calculated F_{P_y} , and MSE for the sand image in Figure 3.5 with $\sigma_{noise} = 100$, actual shift = 0.

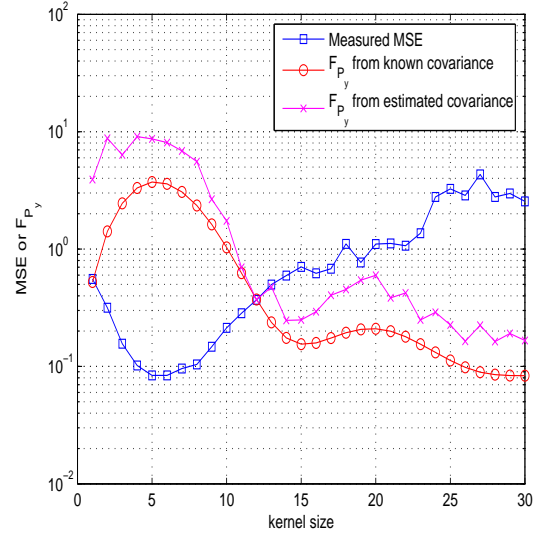


Figure 3.12: Calculated F_{P_y} , and MSE for the water image in Figure 3.7 with $\sigma_{noise} = 100$, actual shift = 0.

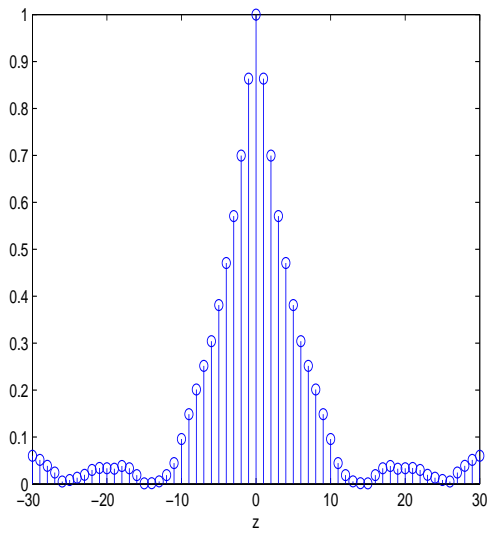


Figure 3.13: Optimal spatial-domain filtering kernel for the Pentagon image in Figure 4.1. This is the Wiener filter which is optimized to minimize the MSE of the filtered image.

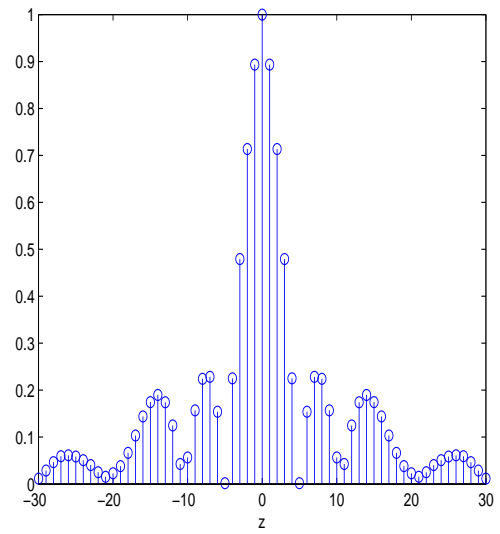


Figure 3.14: Optimal spatial-domain filtering kernel for the Brodatz water image in Figure 3.7. This is the Wiener filter which is optimized to minimize the MSE of the filtered image.



Figure 3.15: 512×512 Tank image from <http://sipi.usc.edu/database/>.

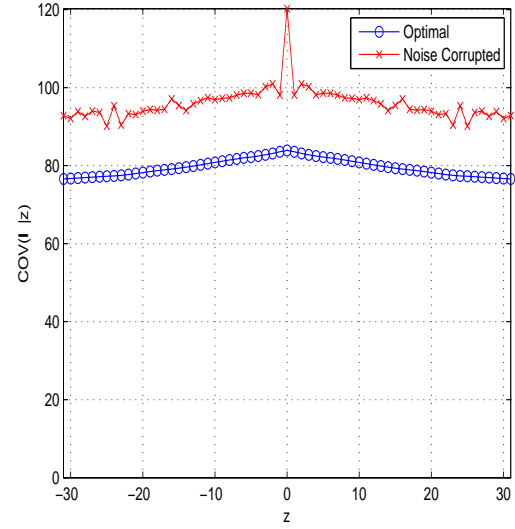


Figure 3.16: Calculated covariance function for the 512×512 Tank image. Note that these covariance functions are more linear than others shown in this chapter.

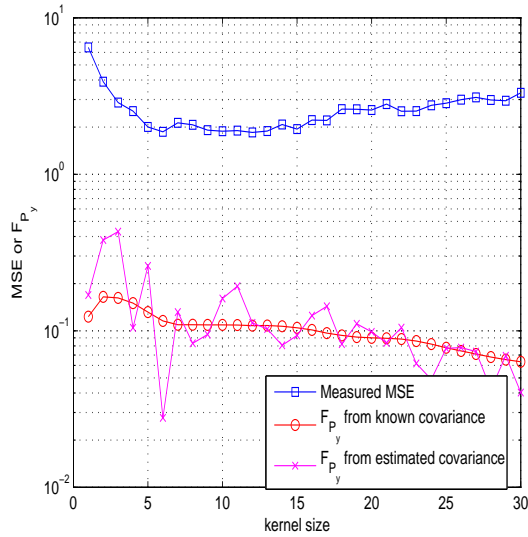


Figure 3.17: Calculated F_{P_y} , and MSE for the tank image in Figure 3.15 with AWGN of $\sigma = 100$.

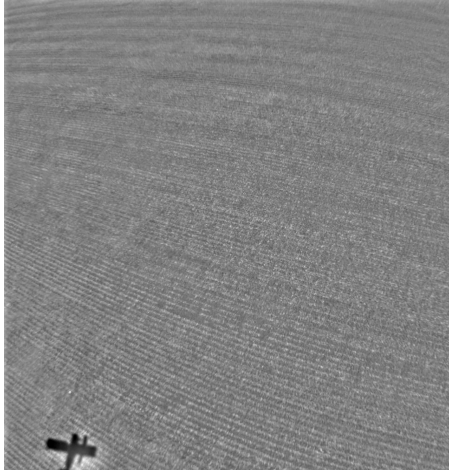


Figure 3.18: Aerial image of cornfield from a sequential series of frames.

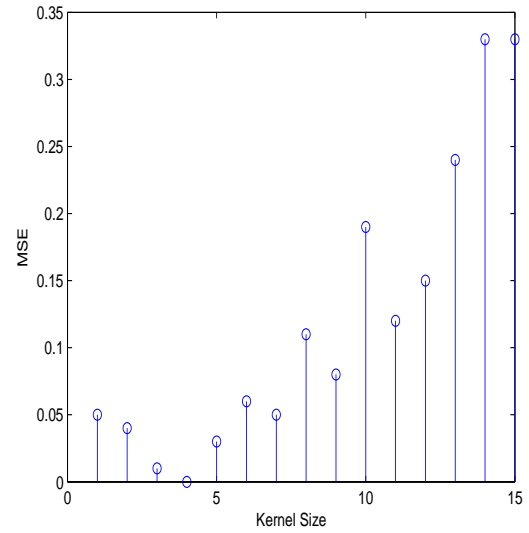


Figure 3.19: Graph of kernel vs MSE for the image of Figure 3.18 with AWGN of $\sigma = 20$ when measuring an actual shift of zero. The optimal kernel size for this image is shown to be four.

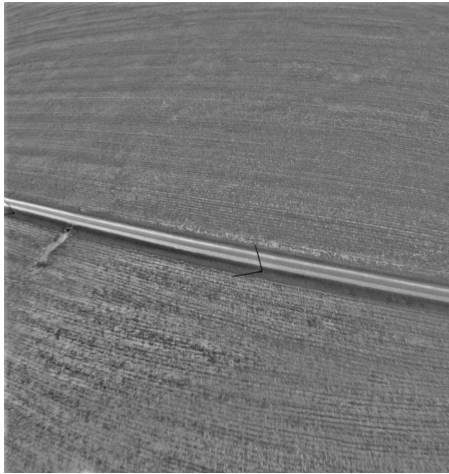


Figure 3.20: An aerial image of a road taken from the series.

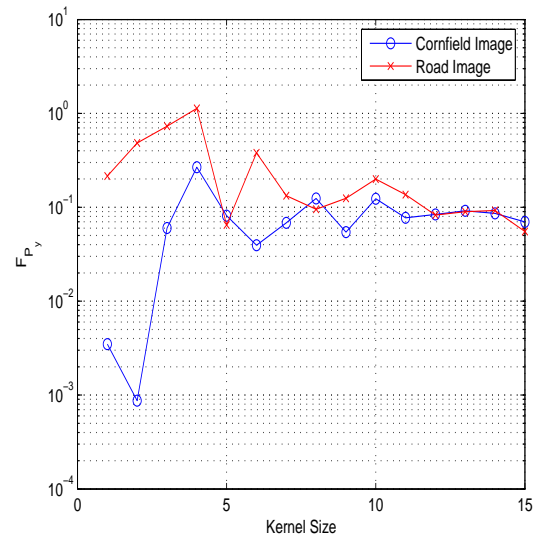


Figure 3.21: Comparison of the calculated F_{P_y} , for the images in Figs. 3.18 and 3.20. Note that both F_{P_y} , peak at Kernel Size = 4.

IV. Bounds on Image Registration Algorithms

This chapter examines the limits on ability to register images using techniques like the one described in Chapter III. This examination is performed using theoretical bounds on image registration algorithms and the effects of filtering on these algorithms. In this chapter, the examination of bounds on image registration performance is extended by applying the Barankin bound to better account for the effects of projecting and filtering images than the CRLB. The chapter also describes bounds on registration in the presence of optical focal-length errors. Bounds on the estimates of the motion of objects pictured within a frame are also discussed since these are an important part of target tracking. Where lengthy derivations are required, these are provided in Appendix A of the dissertation.

4.1 Performance Bounds on Image Registration With Filtered Projections

This section describes the bounds on the registration of a 1-D projection of an image. As described in Chapter III, if an imaging sensor is used primarily for motion estimation, image projections offer what is perhaps the fastest approach for registering available images. The reduction in computational complexity realized by using projections for motion estimation comes at a cost in accuracy. While some literature cites this cost as “minimal” [45], this section provides a derivation of equations that can be used to quantify the theoretical limits on the accuracy of motion estimates derived using projection-based methods. This derivation is similar to the one for the general CRLB found in [29].

The following assumes that images are periodic and band-limited, and that a general filtering kernel \mathbf{H} can be defined which is a positive-definite, circulant matrix. \mathbf{T}_α is circulant and circulant matrices commute, hence

$$(\mathbf{T}_\alpha \mathbf{H} \mathbf{i})(x) = (\mathbf{H} \mathbf{T}_\alpha \mathbf{i})(x) \quad (4.1)$$

An operator \mathcal{D}_z is also defined which works as a convolutional differencing operator across the dimension z . \mathcal{D}_z is also circulant and commutes with \mathbf{T}_α and \mathbf{H} .

The derivation begins by describing the probability distribution function(PDF) of any pixel in the image as [53]

$$p(\mathbf{D}(x, y) | \mathbf{I}(x, y)) = \frac{1}{\sqrt{2\pi}\sigma} \exp \left[-\frac{1}{2\sigma^2} (\mathbf{D}(x, y) - \mathbf{I}(x, y))^2 \right]. \quad (4.2)$$

The PDF of a point in the y projection of the image can be written as

$$p(\mathbf{d}_{1,y}(x) | \mathbf{i}_y(x)) = \frac{1}{\sqrt{2\pi N}\sigma} \exp \left[-\frac{1}{2\sigma^2 N} (\mathbf{d}_{1,y}(x) - \mathbf{i}_y(x))^2 \right], \quad (4.3)$$

and the PDF of the entire projection can be written as

$$p(\mathbf{d}_{1,y} | \mathbf{i}_y) = \left(\frac{1}{\sqrt{2\pi N}\sigma} \right)^N \exp \left[-\frac{1}{2\sigma^2 N} (\mathbf{d}_{1,y} - \mathbf{i}_y)^T (\mathbf{d}_{1,y} - \mathbf{i}_y) \right]. \quad (4.4)$$

If $\mathbf{f}_{i,y}$ is defined to be the projection of an image filtered by \mathbf{H} , the PDF of the filtered projection can be written as the linear transform

$$p(\mathbf{f}_{1,y} | \mathbf{i}_y) = \left(\frac{1}{\sqrt{2\pi N}\sigma} \right)^N \exp \left[-\frac{1}{2\sigma^2 N} (\mathbf{f}_{1,y} - \mathbf{H}\mathbf{i}_y)^T \mathbf{W}^{-1} (\mathbf{f}_{1,y} - \mathbf{H}\mathbf{i}_y) \right], \quad (4.5)$$

where $\sigma^2 N \mathbf{W}$ is the covariance matrix of the projected and filtered noise and $\mathbf{W} = \mathbf{H}\mathbf{H}^T = \mathbf{H}^T \mathbf{H}$. Similarly, if there is a second image which is identical to the first except for a horizontal shift α , the PDF of this projection can be defined as

$$p(\mathbf{f}_{2,y} | \mathbf{i}_y, \alpha) = \left(\frac{1}{\sqrt{2\pi N}\sigma} \right)^N \exp \left[-\frac{1}{2\sigma^2 N} (\mathbf{f}_{2,y} - \mathbf{H}\mathbf{T}_\alpha \mathbf{i}_y)^T \mathbf{W}^{-1} (\mathbf{f}_{2,y} - \mathbf{H}\mathbf{T}_\alpha \mathbf{i}_y) \right]. \quad (4.6)$$

Combining (4.5) and (4.6) to find the joint probability of $\mathbf{f}_{1,y}$ and $\mathbf{f}_{2,y}$ yields

$$\begin{aligned} p(\mathbf{f}_{1,y}, \mathbf{f}_{2,y} | \mathbf{i}_y, \alpha) &= \left(\frac{1}{2\pi N\sigma^2} \right)^N \exp \left[-\frac{1}{2\sigma^2 N} ((\mathbf{f}_{1,y} - \mathbf{H}\mathbf{i}_y)^T \mathbf{W}^{-1} (\mathbf{f}_{1,y} - \mathbf{H}\mathbf{i}_y) \right. \\ &\quad \left. + (\mathbf{f}_{2,y} - \mathbf{H}\mathbf{T}_\alpha \mathbf{i}_y)^T \mathbf{W}^{-1} (\mathbf{f}_{2,y} - \mathbf{H}\mathbf{T}_\alpha \mathbf{i}_y)) \right]. \end{aligned} \quad (4.7)$$

4.1.1 The CRLB of Registration Using Image Projection. The examination on bounds begins with the CRLB. Following the approach used in [48], (4.7) can be used to create a block FIM \mathbf{J} of the form:

$$\mathbf{J} = \begin{pmatrix} \mathbf{J}_{\alpha\alpha} & \mathbf{J}_{\alpha\mathbf{x}}^T \\ \mathbf{J}_{\alpha\mathbf{x}} & \mathbf{J}_{\mathbf{x}\mathbf{x}} \end{pmatrix}. \quad (4.8)$$

In Appendix A.2.1 it is shown that under the assumption of circulant filtering matrices, the Fisher information matrix(FIM) for the filtered projection is identical to the FIM for the original data. Consequently, the FIM can be defined in terms of the unfiltered projections as

$$\mathbf{J}_{\alpha\alpha} = -\mathbb{E} \left[\frac{\partial^2 \ln p(\mathbf{d}_{1,y}, \mathbf{d}_{2,y} | \mathbf{i}_y, \alpha)}{\partial \alpha^2} \right], \quad (4.9)$$

$$\mathbf{J}_{\alpha\mathbf{x}} = -\mathbb{E} \left[\frac{\partial^2 \ln p(\mathbf{d}_{1,y}, \mathbf{d}_{2,y} | \mathbf{i}_y, \alpha)}{\partial \alpha \partial \mathbf{i}_y} \right], \quad (4.10)$$

$$\mathbf{J}_{\mathbf{x}\mathbf{x}} = -\mathbb{E} \left[\frac{\partial^2 \ln p(\mathbf{d}_{1,y}, \mathbf{d}_{2,y} | \mathbf{i}_y, \alpha)}{\partial \mathbf{i}_y^2} \right]. \quad (4.11)$$

As shown in Appendix A.2.1, the FIM can be inverted using block matrix techniques to arrive at the result

$$\text{VAR}(\hat{\alpha}) \geq 2\sigma^2 N [\|\mathcal{D}_x \mathbf{i}_y\|^2]^{-1}. \quad (4.12)$$

This bound derivation using the nuisance parameters differs from the one derived in [46] by a factor of two. This result is apparent for unfiltered images from work in [48] and is suggested by work in [55], but is not explicitly stated in either document.

4.1.2 The Barankin Bound on Registration Using Projections. In the previous subsection, the filtering terms dropped out of the derived bounds. This is also the case for the Barankin bound so the bounds are calculated using the unfiltered PDF of the image. For this case, the FIM calculated using (4.8) is used as \mathbf{J} , the

FIM calculated as in the CRLB, and

$$\begin{aligned} L(\mathbf{d}_1, \mathbf{d}_2 | \boldsymbol{\Theta}_i, \boldsymbol{\Theta}_o) &= \exp \left[\frac{1}{2\sigma^2 N} [(\mathbf{d}_2 - \mathbf{i}_y)^T (\mathbf{d}_2 - \mathbf{i}_y) - (\mathbf{d}_2 - \mathbf{T}_{\alpha_i} \mathbf{i}_y)^T (\mathbf{d}_2 - \mathbf{T}_{\alpha_i} \mathbf{i}_y)] \right], \\ &= \exp \left[\frac{1}{\sigma^2 N} \mathbf{d}_2^T (\mathbf{T}_{\alpha_i} \mathbf{i}_y - \mathbf{i}_y) \right], \end{aligned} \quad (4.13)$$

where $\boldsymbol{\Theta}_o = \mathbf{i}_y$ is the true value of a projected image and $\boldsymbol{\Theta}_i = \mathbf{T}_{\alpha_i} \mathbf{i}_y$ is a projected image shifted by \mathbf{T}_{α_i} . From (4.13), it follows that

$$\begin{aligned} \mathbf{A}_{i,j} &= \mathbb{E} \left[\frac{\partial \ln p(\mathbf{d} | \boldsymbol{\Theta}_o)}{\partial \boldsymbol{\Theta}_o(i)} L(\mathbf{d} | \boldsymbol{\Theta}_j, \boldsymbol{\Theta}_o) \right], \\ &= \begin{cases} \mathbb{E} \left[\frac{1}{\sigma^2 N} (\mathbf{d}_2 - \mathbf{T}_{\alpha_i} \mathbf{i}_y)^T \frac{\partial \mathbf{T}_{\alpha_i} \mathbf{i}_y}{\partial \alpha} \exp \left[\frac{1}{\sigma^2 N} \mathbf{d}_2^T (\mathbf{T}_{\alpha_i} \mathbf{i}_y - \mathbf{i}_y) \right] \right] & : i = 1 \\ \mathbb{E} \left[\frac{1}{\sigma^2 N} [(\mathbf{d}_1(n) - \mathbf{i}_y(n)) + \mathbf{T}_{\alpha} (\mathbf{d}_2(n) - \mathbf{T}_{\alpha_i} \mathbf{i}_y(n)) \mathbf{d}_2^T (\mathbf{T}_{\alpha_i} \mathbf{i}_y - \mathbf{i}_y)] \right] & : i = n + 1, \\ 0 \quad \forall i, j. & \end{cases} \end{aligned} \quad (4.14)$$

The integer shifts of $\alpha_1 = -1$ and $\alpha_2 = 1$ are used as the most likely registration errors and the Barankin bound is calculated using the equations derived in [37] and [38]. The final terms required to calculate the bound for the given conditions are

$$\mathbf{B}_{i,j} = \exp \left[\frac{1}{\sigma^2 N} [(\mathbf{i}_y - \mathbf{T}_{\alpha_i} \mathbf{i}_y)^T (\mathbf{i}_y - \mathbf{T}_{\alpha_j} \mathbf{i}_y)] \right], \quad (4.15)$$

$$\boldsymbol{\Phi} = \begin{pmatrix} -1 & 1 \\ \mathbf{T}_{\alpha_1} \mathbf{i}_y & \mathbf{T}_{\alpha_2} \mathbf{i}_y \end{pmatrix}. \quad (4.16)$$

This yields the following form of the Barankin bound,

$$\mathbb{E}[(\hat{\boldsymbol{\Theta}} - \boldsymbol{\Theta})^2] = \mathbf{J}^{-1} + \boldsymbol{\Phi} \mathbf{B}^{-1} \boldsymbol{\Phi}^T. \quad (4.17)$$

Because the bounds are clearly image dependent, numeric methods are employed. Results calculated using this bound are found in Section 4.3.

4.2 Bounds of Two Dimensional Image Registration With Filtered Images

Image registration using all of the 2-D spatial information available in the image is much more robust than registration using projections alone. This section derives these bounds and also examines the case where a pre-detection image has been corrupted by a focal-length error which is modeled as an optical filter. Consequently, in the two-dimensional case, not only is the derivation slightly more mathematically complicated, but there are also different filtering scenarios that can be accounted for and different bounds that can be examined. In the two-dimensional case, these two significant filtering scenarios are:

1. Optical filtering - filtering is performed in the optics before sensor noise is added and
2. Post-detection filtering - filtering is performed on an image after sensor read-out.

These two cases are derived jointly and the notation \mathbf{H}_p is used to denote post-detection filtering and \mathbf{H}_o is used to denote optical filtering. Since this derivation is similar to the one-dimensional case, it is included as an appendix; however, the results are summarized in the following subsections.

4.2.1 2-D CRLB with Optical Filtering. The registration of two frames of data, \mathbf{D}_1 and \mathbf{D}_2 as defined in (1.8) and (1.9) is examined first. The noise in these frames (\mathbf{Q}_1 and \mathbf{Q}_2) is assumed to be spatially and temporally uncorrelated and Gaussian. This noise model typifies fixed pattern noise and read noise in the readout amplifier of a CCD that are typically the dominant noise sources in very-low-intensity images [20], [25]. The natural logarithm of the joint probability of these

two frames represented as vectors (1.8) and (1.9) can be written as

$$\begin{aligned}\ln p(\mathbf{D}_1, \mathbf{D}_2 | \mathbf{I}) &= -\frac{1}{2\pi\sigma^2}(\mathbf{D}_1 - \mathbf{H}_p \mathbf{H}_o \mathbf{I})^T \mathbf{W}^{-1}(\mathbf{D}_1 - \mathbf{H}_p \mathbf{H}_o \mathbf{I}) \\ &\quad -\frac{1}{2\pi\sigma^2}(\mathbf{D}_2 - \mathbf{H}_p \mathbf{H}_o \mathbf{T}_{\alpha,\beta} \mathbf{I})^T \mathbf{W}^{-1}(\mathbf{D}_2 - \mathbf{H}_p \mathbf{H}_o \mathbf{T}_{\alpha,\beta} \mathbf{I}) \\ &\quad + \text{constant},\end{aligned}\tag{4.18}$$

where now $\mathbf{W} = \mathbf{H}_p \mathbf{H}_p^T = \mathbf{H}_p^T \mathbf{H}_p$. Using the derivation found in Appendix Section A.2.2, the CRLB for the 2-D case is found to be

$$\text{VAR}(\hat{\alpha}) \geq \frac{2\sigma^2 \|\mathcal{D}_y \mathbf{H}_o \mathbf{I}\|^2}{\|\mathcal{D}_x \mathbf{H}_o \mathbf{I}\|^2 \|\mathcal{D}_y \mathbf{H}_o \mathbf{I}\|^2 - \langle \mathcal{D}_x \mathbf{H}_o \mathbf{I}, \mathcal{D}_y \mathbf{H}_o \mathbf{I} \rangle^2},\tag{4.19}$$

$$\text{VAR}(\hat{\beta}) \geq \frac{2\sigma^2 \|\mathcal{D}_x \mathbf{H}_o \mathbf{I}\|^2}{\|\mathcal{D}_x \mathbf{H}_o \mathbf{I}\|^2 \|\mathcal{D}_y \mathbf{H}_o \mathbf{I}\|^2 - \langle \mathcal{D}_x \mathbf{H}_o \mathbf{I}, \mathcal{D}_y \mathbf{H}_o \mathbf{I} \rangle^2},\tag{4.20}$$

where $\|\cdot\|^2$ is the square of the L2 norm of a vector, $\langle \cdot, \cdot \rangle$ is the inner product of two vectors, and \mathbf{I} is a vectorized version of a 2-D image. In this case, as the size of the filtering kernel increases, the magnitude of the terms $\|\mathcal{D}_x \mathbf{H}_o \mathbf{I}\|^2$ and $\|\mathcal{D}_y \mathbf{H}_o \mathbf{I}\|^2$ decreases. Intuitively, as resolution is lost in the image, it becomes more difficult to register. This, however, is not the case for filtering performed after detection since \mathbf{H}_p has dropped out of (4.19) and (4.20). It is interesting to note that although filtering images has been shown to improve the performance of correlation and gradient-based image registration [4], [39], [46], [47], the type of post-detection filtering performed in these papers does not improve the CRLB. Rather, post-detection filtering is a part of the estimation process that may yield performance approaching the bound.

4.2.2 2-D Barankin Bound. Looking at the two-dimensional case for optical filtering, \mathbf{J} is again the $(N+2) \times (N+2)$ FIM as derived for the CRLB. If the most likely errors are expanded to include shift vectors of

$$[\alpha_i, \beta_i] \in \{[1 \ 0], [-1 \ 0], [0 \ 1], [0 \ -1]\},$$

then the matrix Φ can be constructed as

$$\Phi = \begin{pmatrix} -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ \mathbf{T}_{\alpha,\beta}\mathbf{H}_o\mathbf{I}|_{\alpha=-1,\beta=0} & \mathbf{T}_{\alpha,\beta}\mathbf{H}_o\mathbf{I}|_{\alpha=1,\beta=0} & \mathbf{T}_{\alpha,\beta}\mathbf{H}_o\mathbf{I}|_{\alpha=0,\beta=-1} & \mathbf{T}_{\alpha,\beta}\mathbf{H}_o\mathbf{I}|_{\alpha=0,\beta=1} \end{pmatrix}. \quad (4.21)$$

In this case the likelihood function is calculated as

$$\begin{aligned} L(\mathbf{D}_1, \mathbf{D}_2 | \Theta_i, \Theta_o) &= \frac{p(\mathbf{D}_1, \mathbf{D}_2 | \mathbf{H}_o \mathbf{T}_{\alpha,\beta} \mathbf{I}, \alpha_i, \beta_i)}{p(\mathbf{D}_1, \mathbf{D}_2 | \mathbf{H}_o \mathbf{T}_{\alpha,\beta} \mathbf{I}, \alpha = 0, \beta = 0)} \\ &= \exp \left[\frac{1}{\sigma^2} \mathbf{D}_2^T (\mathbf{H}_o \mathbf{T}_{\alpha_i, \beta_i} \mathbf{I} - \mathbf{H}_o \mathbf{I}) \right]. \end{aligned} \quad (4.22)$$

Beginning with (A.20), the partial derivatives of the log-likelihood functions are

$$\begin{aligned} \frac{\partial \ln p(\mathbf{D}_1, \mathbf{D}_2 | \mathbf{I}, \alpha, \beta)}{\partial \alpha} &= \frac{1}{\sigma^2} (\mathbf{D}_2 - \mathbf{H}_o \mathbf{T}_{\alpha,\beta} \mathbf{I})^T \frac{\partial \mathbf{H}_o \mathbf{T}_{\alpha,\beta} \mathbf{I}}{\partial \alpha}, \\ \frac{\partial \ln p(\mathbf{D}_1, \mathbf{D}_2 | \mathbf{I}, \alpha, \beta)}{\partial \beta} &= \frac{1}{\sigma^2} (\mathbf{D}_2 - \mathbf{H}_o \mathbf{T}_{\alpha,\beta} \mathbf{I})^T \frac{\partial \mathbf{H}_o \mathbf{T}_{\alpha,\beta} \mathbf{I}}{\partial \beta}, \\ \frac{\partial \ln p(\mathbf{D}_1, \mathbf{D}_2 | \mathbf{I}, \alpha, \beta)}{\partial \mathbf{I}} &= \frac{1}{\sigma^2} [\mathbf{H}^T (\mathbf{D}_1 - \mathbf{H}_o \mathbf{I}) + (\mathbf{T}_{\alpha,\beta} \mathbf{H}_o)^T (\mathbf{D}_2 - \mathbf{H}_o \mathbf{T}_{\alpha,\beta} \mathbf{I})]. \end{aligned}$$

It is easy to see that, as with the one dimensional case, $\mathbf{A}_{i,j} = 0 \ \forall \ i, j$. Finally,

$$\mathbf{B}_{i,j} = \exp \left[\frac{1}{\sigma^2} \left[(\mathbf{H}_o \mathbf{I} - \mathbf{H}_o \mathbf{T}_{\alpha_i, \beta_i} \mathbf{I})^T (\mathbf{H}_o \mathbf{I} - \mathbf{H}_o \mathbf{T}_{\alpha_j, \beta_j} \mathbf{I}) \right] \right], \quad (4.23)$$

with which the Barankin bound can be calculated numerically as a function of the CRLB from (A.34),

$$\mathbb{E}[(\hat{\Theta} - \Theta)^2] \geq \mathbf{J}^{-1} + \Phi \mathbf{B}^{-1} \Phi^T. \quad (4.24)$$

Results for optically-filtered images calculated using this bound are found in Section 4.3.

4.3 Experimental Results

To examine the effect of the calculated bounds on typical images, bounds were calculated and examined for a standard image and for an ensemble of frames of LIDAR data. Bounds were calculated numerically for different image sizes, different image intensities and for varying registration methods. Bounds were specifically examined for the registration of images when using projections of the images and for the more robust case of 2-D image registration. The effects of focal-length errors on the CRLB and the Barankin bound for the images were also examined. The results obtained show that small image size, low illumination intensity, and focal-length errors increase the relevance of the Barankin bound to registration estimates.

In the following discussion and graphs, 8-bit gray-scale images are employed and the degree of noise in an image is measured using PSNR where the PSNR of an $S \times T$ sized image is defined as:

$$PSNR = 10 \log_{10} \left(\frac{255^2}{\frac{1}{ST} \|\mathbf{I} - \hat{\mathbf{I}}\|_F^2} \right), \quad (4.25)$$

where \mathbf{I} is the database image, $\hat{\mathbf{I}}$ is the corrupted version of the image, and $\|\dots\|_F$ is the Frobenius norm which is defined as the square root of the sum of the squares of the elements of a matrix. For experimental LIDAR images, where images intensities are measured by photon counts, the maximum value of the average frame is used in place of 255 in the numerator of (4.25).

4.3.1 Registration Performance for Standard Pentagon Image. To simulate low-SNR conditions that might be present in night-time or passive infrared (PIR) filtering, the intensity of the image shown in Figure 4.1 was divided by 4 and then corrupted with AWGN. For the original images, this produced pixel values in the range [15, 60] where the maximum possible pixel value was 255.



Figure 4.1: 1024×1024 Pentagon image from <http://sipi.usc.edu/database/>.

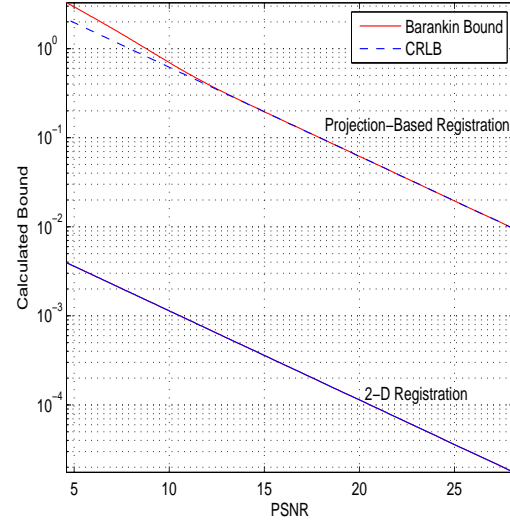


Figure 4.2: Bound on the variance of estimates of the x -shift for the image shown in Figure 4.1 using both projections and 2-D registration. Bounds for 2-D registration are lower than those derived using projections. To simulate low-light conditions, the pixel values of the source image are divided by a factor of 4.

4.3.1.1 Performance of Projection and Two-Dimensional Registration.

The Barankin bound and the CRLB were first calculated for the full image and for a 128×128 subsection of the image shown in Figure 4.3, without any defocus errors (i.e. $\mathbf{H}_0 = \mathcal{I}$). For both the 1024×1024 image, and the 128×128 subsection, the CRLB and the Barankin bound of 2-D horizontal shift estimates were the same, for practical purposes. With less information, it was expected that the Barankin bound would be more pronounced for the smaller image of Figure 4.3 and for the projection bounds in general. This was, in fact, the case as shown by examining and comparing the results of the analytical bound calculation shown in Figures 4.2 and 4.4. In both cases, the bound on estimates using projections is higher than those on 2-D estimates; however, the breakpoint of the Barankin bound occurs at the highest



Figure 4.3: 128×128 subsection of Figure 4.1.

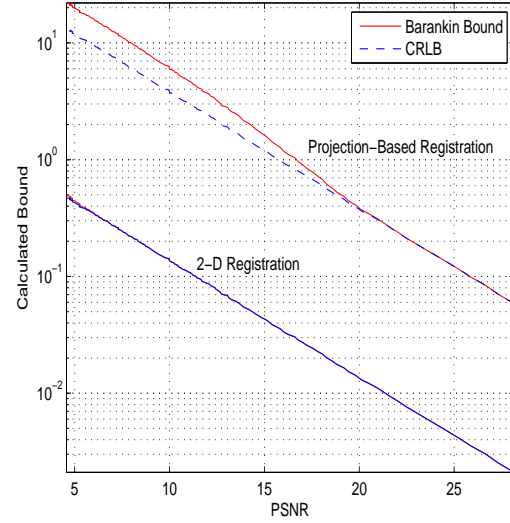


Figure 4.4: Calculated performance bounds for registration using projections and 2-D registration for the 128×128 subsection of the Pentagon image in Figure 4.3 where the intensities of the source images have been divided by a factor of 4.

SNR in the case of estimates for Figure 4.3. In general, the bounds on registration using two-dimensional correlation were much lower than the bounds on registration using 1-D projections and the deviation of the Barankin bound from the CRLB is much less pronounced in 2-D filtering.

4.3.1.2 Bounds on Registration in the Presence of Defocus Errors.

Bounds on registration estimates were then calculated using the images of Figures 4.2 and 4.4. This required making some basic assumptions about the optical system under study. The calculated bounds for the optical filtering case were performed using an optical model based on specifications from a Celestron 14" (356 mm) Schmidt-Cassegrain telescope operated at a range-to-objective of 20 km. This telescope has a focal length of approximately four meters. Calculations were performed for light with a wavelength of $0.5 \mu m$. Measurements were then simulated of a diffraction-



Figure 4.5: Image shown in Figure 4.1 with simulated 0.7λ defocus error.

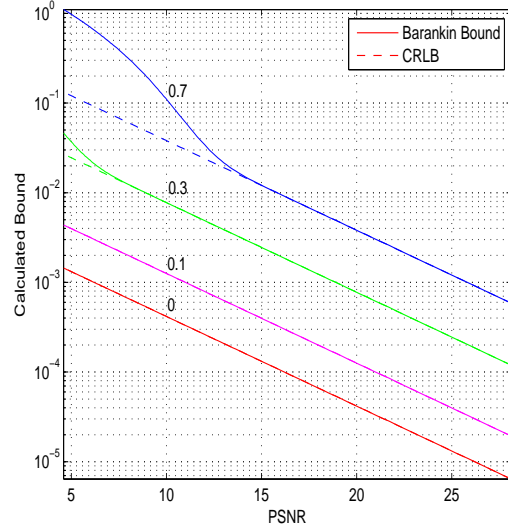


Figure 4.6: Bounds for the image shown in Figure 4.1 with simulated 0, 0.1λ , 0.3λ , and 0.7λ defocus errors. Note that the difference between the bounds increases with increasing defocus.

limited image with focal-length errors of 0 to 0.7 wavelengths in increments of 0.1 wavelengths. Representative defocused images are shown in Figures 4.5 and 4.7.

As shown in Figures 4.6 and 4.8, the bounds on shift estimates for both images increased with increasing defocus error. Similarly, the Barankin bound became increasingly relevant with increased defocus errors - especially in the case of the smaller images. As before, the smaller image had higher overall bounds due to decreased information content.

4.3.2 Registration Performance of Actual LIDAR data. Using insights gained from the examination of standard test images, a series of 50 frames of LIDAR data captured using techniques and equipment described in [36] was examined. These individual frames were median filtered to remove specular returns and spatially registered using a two-dimensional cross-correlation. Then after filtering and

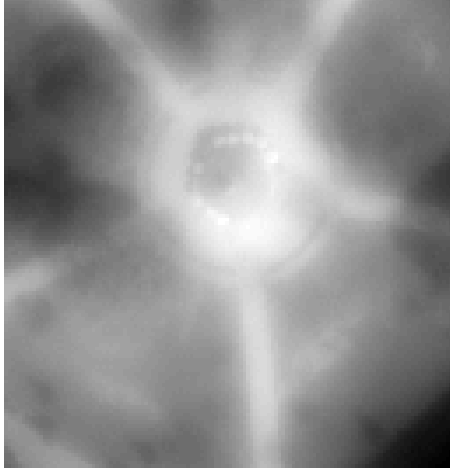


Figure 4.7: Image shown in Figure 4.3 with simulated 0.7λ defocus error.

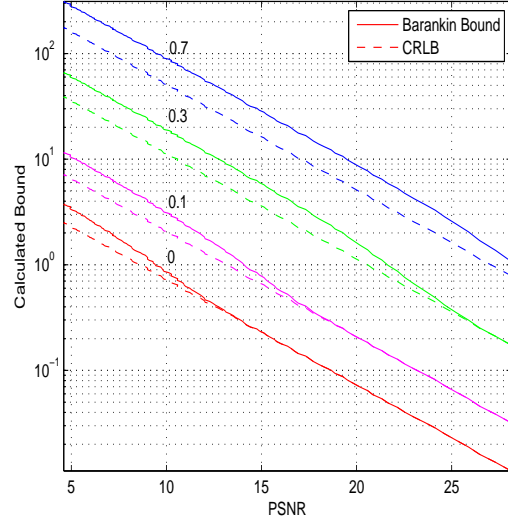


Figure 4.8: Bounds for the image shown in Figure 4.3 with simulated 0, 0.1λ , 0.3λ , and 0.7λ defocus errors. Note that smaller image results in bounds higher than those displayed Figure 4.6 and that the difference between the bounds increases with increasing defocus.

registration, the frames were averaged to create a representative 256×256 diffraction-limited image which was considered to be “truth” data. This resulting image is shown in Figure 4.9 and a representative frame of data is shown in Fig. 4.10. Of particular interest was the region of interest shown in Figure 4.11.

Using the frame average in conjunction with registration estimates of the individual frames, PSNRs were calculated for each of the 50 frames of data. The calculated PSNRs for the frames ranged from 25.7 dB to 27.7 dB with a mean dB value of 27.07 dB.

The frame average was also used to calculate theoretical bounds on the MSE of registration estimates for the frames in the ensemble. Bounds on estimates of column shifts of the frames using both projections and 2-D estimates for the full

		Projection-Based Registration		2-D Registration	
	Est PSNR	Cramer-Rao	Barankin	Cramer-Rao	Barankin
Full Image	26.3 dB	0.153	0.153	0.0089	0.0089
Region of Support	24.6 dB	0.790	0.953	0.0471	0.0471

Table 4.1: Calculated registration bounds for full frame and region of interest of LIDAR data in units of pixels².

image are shown graphically in Figure 4.13. The threshold for the Barankin bound (the PSNR below which the bound diverges from the CRLB) is shown to occur at an approximate PSNR of 23.5. The lowest PSNR was 25.7 dB which lies in a region where the CRLB and the Barankin bound are coincident. Thus, for the full frames, the CRLB is an adequate measure of the bound on projection-based registration. Bounds were also calculated for registration estimates performed using 2-D shift estimation algorithms. As expected, bounds on variances on these estimates were significantly lower than those calculated using projection-based methods. 2-D bounds are also shown graphically in Figure 4.13. A comparison of the CRLB and Barankin bounds for 1-D and 2-D image registration given the calculated PSNRs of the LIDAR data is shown in Table 4.1.

For automatic target recognition problems, it is often necessary to identify and estimate motion in a specific target among background clutter. For these applications, the ability to estimate motion of an object is also theoretically bounded. Examination of these bounds began by selecting a subregion of interest within the image shown in Figure 4.9. The target for this experiment was a 68×168 region of interest shown in Figure 4.11. An estimate of the motion of the tank between consecutive frames depends on the ability to register the regions of interest in consecutive frames. As shown in Figure 4.14, the bounds on registration with projections are significantly higher than with 2-D registration techniques. It is also interesting to note that the Barankin bound for registration using projections is approximately 30% higher than the CRLB within the range of PSNRs encountered in the data.

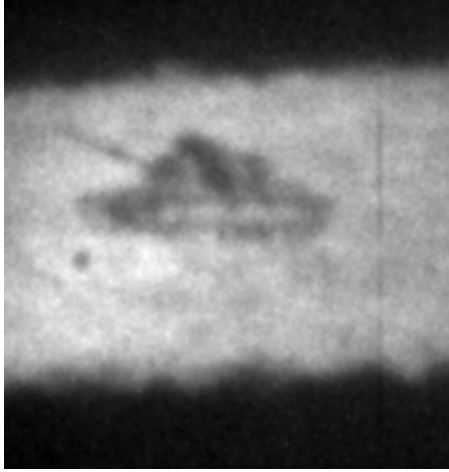


Figure 4.9: 256×256 image resulting from median filtering, and averaging 50 frames of LIDAR data captured at 10 km from the target. “Truth” Data (approx.)

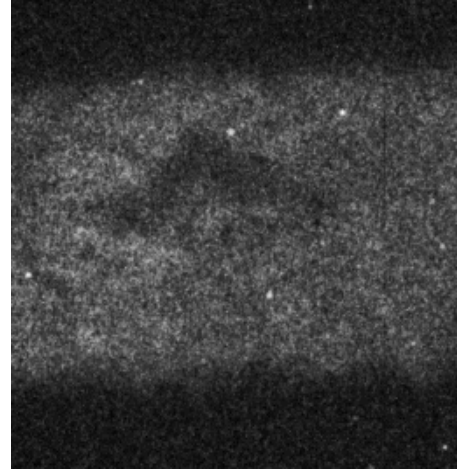


Figure 4.10: Representative LIDAR frame prior to filtering and averaging (PSNR = 26.3).

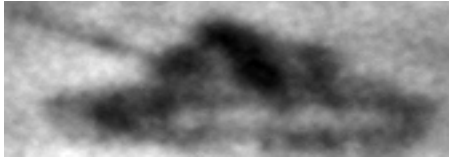


Figure 4.11: 68×168 region of interest within the image of Figure 4.9.

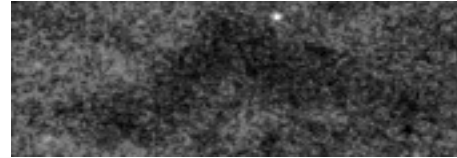


Figure 4.12: Representative region of interest in a LIDAR frame prior to filtering and averaging (PSNR = 24.6).

4.4 Chapter Summary

This chapter provided a calculation and comparison of theoretical performance bounds for image registration algorithms. It showed that for large images under conditions of full-frame registration, the CRLB is an adequate measure of performance for most realistic imaging conditions. For projected, small images, or image corrupted by focal-length errors, however, the CRLB may not sufficiently predict bounds on the performance of a registration algorithm and the Barankin bound provides a more accurate estimate.

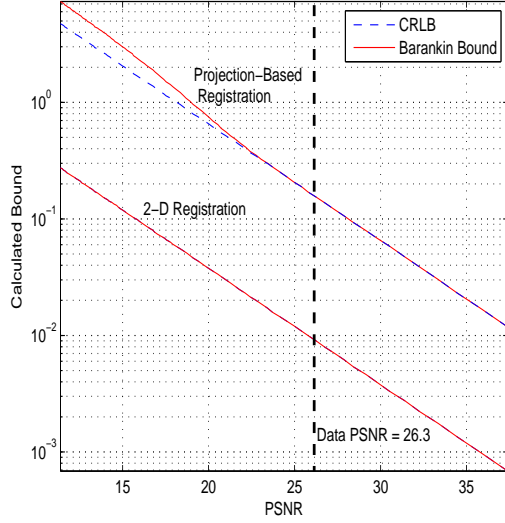


Figure 4.13: Bounds on registration using projections and 2-D registration of the LIDAR frame shown in Figure 4.9. The CRLB and Barankin bound for the 2-D registration case are indistinguishable.

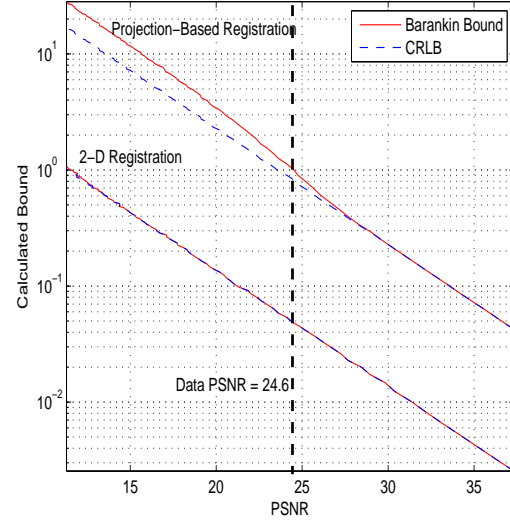


Figure 4.14: Bounds on registration using projections of the LIDAR frame region of interest shown in Figure 4.11. For projection-based registration, the breakpoint of the Barankin bound is approximately 29.0 which is well above the PSNRs of the region of interest. In the 2-D registration, the breakpoint of the Barankin bound falls far below the average PSNRs for the data. This indicates that the CRLB is an adequate bound for 2-D but not 1-D registration of this data set.

This chapter also showed that understanding of registration bounds is extended by calculating the Barankin bound. Calculations for the Barankin bound were largely numeric and were based on the most probable registration errors. For the images examined, image registration using projections increased both the CRLB and the Barankin bound as compared to bounds derived using 2-D registration algorithms. However, it is also worth noting that Cain *et al.* [11] show that shift estimates for low intensity images in the presence of fixed pattern noise may actually be better using projections than full 2-D estimates. Bounds on this behavior would also make an interesting future study.

It was also shown that for the test images, the CRLB and Barankin bounds increased as the severity of the defocus errors increased. With this increasing defocus, it was demonstrated that the Barankin bound became more pronounced and more applicable to images with higher SNRs.

Perhaps the most interesting aspect of the research documented in this chapter was that the bounds under study were most applicable to distortions of small images. In many target-recognition applications, objects being imaged may be rotated and dilated and salient features may be extracted using various filtering techniques. This observation suggests many follow-on applications. For instance, the Barankin bound may be of increasing importance to applications where images are aliased or in the differentiation of multiple similar targets. Another interesting extension would be the calculation of bounds on identification of objects and features with contrasting colors under low light conditions. Other extensions to this research will be discussed in Chapter VI of this dissertation.

V. *Block-based Methods for Denoising Images*

Block-based denoising algorithms approach single-frame denoising using techniques that are similar to the multiframe averaging techniques facilitated by image registration. This chapter introduces several new block-based denoising algorithms that produce impressive results, especially in low-SNR scenarios.

The methods that are presented in this chapter begin by thresholding the variance of individual blocks to identify areas that are effectively handled by standard image processing techniques. Like the method proposed by Kervrann and Boulanger [31], the methods described here use the Euclidian distance between blocks and develop a threshold based on the chi-square distribution to identify matching blocks. Unlike the method proposed by Kervrann and Boulanger, the methods used in this chapter use fast approximations for determining these thresholds, examine block correlations and higher order statistics of the error function to match blocks with similar content, and rely on a simple binary weighting scheme to combine blocks in a way that produces a denoised estimate of a region of interest. These new methods also improve on low-SNR performance of other methods by suspending the requirement for comparative blocks to be spatially close. This allows combining data from across an entire image and, in fact, could facilitate combining data from entirely different image frames from the same sensor.

5.1 *The Gaussian Detection Denoising Method*

Using the mathematical background described in Appendix B, the NLM algorithm was modified to improve its performance. This section describes how the new algorithm was formulated and implemented to improve the performance of the NLM algorithm. The method exploits redundancy in the image and improves on both the theoretical foundation and the output of the NLM algorithm. This new algorithm which will heretofore be known as the “Gaussian Detection Denoising”

or GDD method. The algorithm is briefly described below and will be explained in detail later in this section.

5.1.1 Overview of the GDD Denoising Method.

1. Begin with an image of size $S \times T$.
2. Select an $N \times N$ -sized subimage centered at (i, j) within the image to denoise. Call this block $\mathbf{F}_{i,j}$.
3. For every pixel centered at (s, t) in the parent image, define a neighborhood around the pixel and call this neighborhood the block $\mathbf{G}_{s,t}$.
4. Subtract the means from $\mathbf{F}_{i,j}$ and all $\mathbf{G}_{s,t}$ blocks.
5. For all $\mathbf{G}_{s,t}$ calculate the scalar $\mathbf{A}(s, t)$ that minimizes the mean-squared error between $\mathbf{F}_{i,j}$ and $\mathbf{A}(s, t)\mathbf{G}_{s,t}$.
6. For all $\mathbf{G}_{s,t}$ calculate the mean square of $\mathbf{F}_{i,j} - \mathbf{A}(s, t)\mathbf{G}_{s,t}$. Call this value $MSE(s, t)$.
7. Examine the distribution of $MSE(s, t)$ for all s and t . Determine if it has a Gaussian distribution or if it is possible to detect a subset of MSEs that naturally form a Gaussian distribution.
8. If no Gaussian exists, use the original value of $\mathbf{F}_{i,j}$ (not just the center pixel) as the denoised value.
9. If a Gaussian does exist, average the values of $\mathbf{F}_{i,j}$ and the $\mathbf{G}_{s,t}$ blocks that form the Gaussian. Call this averaged block the new value for $\mathbf{F}_{i,j}$.
10. Restore the mean to the block $\mathbf{F}_{i,j}$. This is the denoised value of $\mathbf{F}_{i,j}$.
11. Repeat steps 2-10 for other blocks in the image as desired.
12. Recombine the individual blocks $\mathbf{F}_{i,j}$.

5.1.2 *GDD Preliminary Assumptions and Calculations.* To consider the algorithm in more detail, suppose that there exists an image \mathbf{D} , a region of interest defined $\mathbf{F}_{i,j}$, and subimages used for comparison defined as $\mathbf{G}_{s,t}$ as in Section 2.3. Furthermore, call the index of each pixel in these subimages (u, v) where $u \in \mathcal{U} \triangleq \{1, \dots, \mathcal{N}\}$ and $v \in \mathcal{V} \triangleq \{1, \dots, \mathcal{N}\}$. To remove illumination and reflectance differences, the means are removed from both the $\mathbf{F}_{i,j}$ and $\mathbf{G}_{s,t}$. That is, for all values of \mathbf{F} and \mathbf{G} indexed by x and y :

$$\hat{\mathbf{F}}_{i,j}(x, y) = \mathbf{F}_{i,j}(x, y) - \frac{1}{N^2} \sum_{u \in \mathcal{U}, v \in \mathcal{V}} \mathbf{F}_{i,j}(u, v), \quad (5.1)$$

and

$$\hat{\mathbf{G}}_{s,t}(x, y) = \mathbf{G}_{s,t}(x, y) - \frac{1}{N^2} \sum_{u \in \mathcal{U}, v \in \mathcal{V}} \mathbf{G}_{s,t}(u, v) \quad \forall \mathbf{G}_{s,t} \in \mathbf{D}. \quad (5.2)$$

Then define

$$\text{GDD}(\hat{\mathbf{F}}_{i,j}) = \frac{\sum_{s \in \mathcal{S}, t \in \mathcal{T}} \mathbf{A}(s, t) \hat{\mathbf{G}}_{s,t}}{\sum_{s \in \mathcal{S}, t \in \mathcal{T}} \mathbf{A}(s, t)}. \quad (5.3)$$

It is desirable to use weights that are not necessarily dependent on *a priori* knowledge of the noise. Instead of attempting to calculate a unique weight for each $\hat{\mathbf{G}}_{s,t}$, two hypotheses are posed:

$$H^1 : \left\| \hat{\mathbf{F}}_{i,j} - \hat{\mathbf{G}}_{s,t} \right\|_F^2 \approx \left\| \mathbf{Q}_{i,j} - \mathbf{Q}_{s,t} \right\|_F^2, \quad (5.4)$$

$$H^0 : \left\| \hat{\mathbf{F}}_{i,j} - \hat{\mathbf{G}}_{s,t} \right\|_F^2 \not\approx \left\| \mathbf{Q}_{i,j} - \mathbf{Q}_{s,t} \right\|_F^2. \quad (5.5)$$

In (5.4) and (5.5), $\mathbf{Q}_{i,j}$ and $\mathbf{Q}_{s,t}$ are realizations of the noise in $\hat{\mathbf{G}}_{s,t}$ and $\hat{\mathbf{F}}_{i,j}$. In other words, on average, H^1 corresponds to the case when $\hat{\mathbf{G}}_{s,t}$ and $\hat{\mathbf{F}}_{i,j}$ are approximately

identical but differ by AWGN and H^0 corresponds to the case when the difference between $\hat{\mathbf{G}}_{s,t}$ and $\hat{\mathbf{F}}_{i,j}$ is greater than that attributable to noise.

The problem with using these hypotheses is that the only available data is on the left hand sides of (5.4) and (5.5) and consequently, it is not possible to develop a traditional likelihood ratio test. However, using the background provided in Appendix B, it is possible to develop an alternate test which allows a rough differentiation between these two cases. As discussed in Appendix B, if there exist a $\hat{\mathbf{G}}_{s,t}$ such that $\left\| \hat{\mathbf{F}}_{i,j} - \hat{\mathbf{G}}_{s,t} \right\|_F^2 \approx \left\| \mathbf{Q}_{i,j} - \mathbf{Q}_{s,t} \right\|_F^2$, then the distribution of the MSEs of $\left\| \hat{\mathbf{F}}_{i,j} - \hat{\mathbf{G}}_{s,t} \right\|_F^2$ will be a noncentral chi-square distribution. Without *a priori* knowledge of the distribution of the noise it is not possible to estimate λ ; however, ν corresponds to the number of pixels in $\mathbf{F}_{i,j}$ and $\mathbf{G}_{s,t}$. As ν increases towards infinity, the shape of the distribution $\chi_\nu^2(\lambda)$ becomes Gaussian as demonstrated graphically in Figure 5.1. Therefore, if ν is chosen to be sufficiently large, it is expected that when $\mathbf{F}_{i,j}$ and $\mathbf{G}_{s,t}$ are approximately equal except for additive noise, the distribution of MSEs between $\mathbf{F}_{i,j}$ and $\mathbf{G}_{s,t}$ will be Gaussian. Therefore, the two hypotheses are reposed as follows:

$$H^1 : \left\| \hat{\mathbf{F}}_{i,j} - \hat{\mathbf{G}}_{s,t} \right\|_F^2 \in \left\{ \left\| \hat{\mathbf{F}}_{i,j} - \hat{\mathbf{G}}_{s,t} \right\|_F^2 \mid \left\| \hat{\mathbf{F}}_{i,j} - \hat{\mathbf{G}}_{s,t} \right\|_F^2 \text{ forms a Gaussian distribution} \right\} \quad (5.6)$$

$$H^0 : \text{else.} \quad (5.7)$$

As described in more detail in Section 5.1.3, the GDD algorithm will determines this hypothesis by iteratively setting a threshold within the MSEs of each block $\mathbf{F}_{i,j}$ and testing to see whether the distribution of MSEs below this threshold is Gaussian. When $\hat{\mathbf{G}}_{s,t} \equiv \hat{\mathbf{F}}_{i,j}$, it also assumes H^1 . Using the two hypotheses, it is then possible

to assign a binary weight to $\mathbf{A}(s, t)$ such that

$$H^1 \Rightarrow \mathbf{A}(s, t) = 1,$$

$$H^0 \Rightarrow \mathbf{A}(s, t) = 0.$$

5.1.3 Observed Distribution of Mean Squared Errors. For a given image \mathbf{D} , examining the histogram of the MSEs between a single $\hat{\mathbf{F}}_{i,j}$ and all other $\hat{\mathbf{G}}_{s,t}$, yields something that may be similar in appearance to Figure 5.2. As noted, when $\hat{\mathbf{F}}_{i,j}$ and $\hat{\mathbf{G}}_{s,t}$ are sufficiently large, if there is sufficient redundancy in the image, the distribution of the MSEs will be a combination of two distributions, one of which is a $\chi^2_\nu(\lambda)$ distribution that approximates a Gaussian. This is demonstrated in Figure 5.1. If the variance of the noise is known, the λ and the location of $E[\chi^2_\nu(\lambda)]$ can be estimated. In the absence of this knowledge, it may be possible to locate a Gaussian distribution in the distribution of the Mean Squared Errors by scaling the MSEs and using an iterative process to detect a Gaussian distribution. The subimages, $\hat{\mathbf{G}}_{s,t}$, with MSEs that lie in this Gaussian distribution are considered as satisfying H^1 .

5.1.3.1 Detection of H^1 and H^0 . To perform this separation, first find a constant $\mathbf{A}(s, t)$ for each $\hat{\mathbf{G}}_{s,t}$ that minimizes the MSE between $\hat{\mathbf{F}}_{i,j}$ and $\mathbf{A}(s, t)\hat{\mathbf{G}}_{s,t}$. This can be accomplished by minimizing the quantity

$$MSE = \frac{1}{N^2} \left\| \hat{\mathbf{F}}_{i,j} - \mathbf{A}(s, t)\hat{\mathbf{G}}_{s,t} \right\|_F^2.$$

When this equation is expanded out, it yields the following where the notation $\langle \mathbf{F}, \mathbf{G} \rangle$ indicates an inner product and $\Re\{\}$ indicates taking the real part of a complex number

$$MSE = \frac{1}{N^2} \left(\left\| \hat{\mathbf{F}}_{i,j} \right\|_F^2 - 2\mathbf{A}(s, t)\Re\left\{ \left\langle \hat{\mathbf{F}}_{i,j}, \hat{\mathbf{G}}_{s,t} \right\rangle \right\} + \mathbf{A}(s, t)^2 \left\| \hat{\mathbf{G}}_{s,t} \right\|_F^2 \right).$$

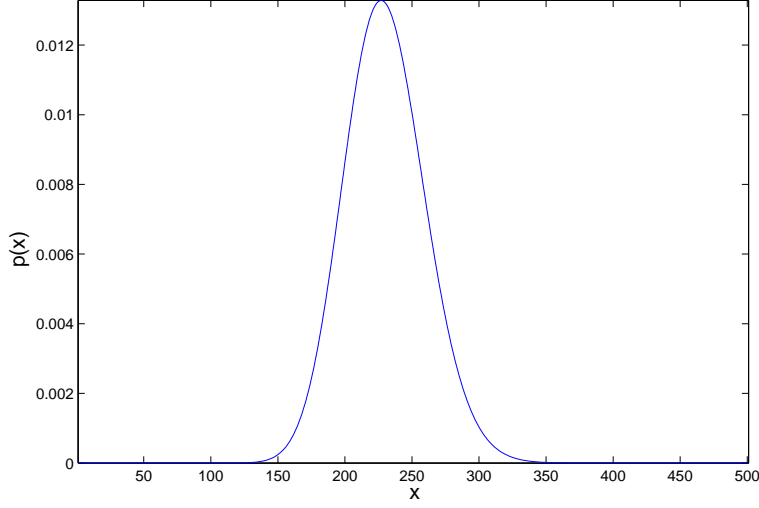


Figure 5.1: Graph of the PDF $\chi_{225}^{\prime 2}(0)$ over $0 \leq x \leq 500$ which occurs for $N = 15$.

For intensity images (i.e. $\mathbf{I} = \Re\{\mathbf{I}\}$), setting the first derivative of this equation equal to zero and solving for $\mathbf{A}(s, t)$ produces

$$\mathbf{A}(s, t) = \frac{\langle \hat{\mathbf{F}}_{i,j}, \hat{\mathbf{G}}_{s,t} \rangle}{\|\hat{\mathbf{G}}_{s,t}\|_F^2}. \quad (5.8)$$

If the histogram resulting from the calculation of MSE is examined for all values of $\mathbf{A}(s, t)$ and $\hat{\mathbf{G}}_{s,t}$, a distribution is arrived at that appears similar to Figure 5.3 which has an upper bound of $\|\hat{\mathbf{F}}_{i,j}\|_F^2 / N^2$. Subimages $\hat{\mathbf{G}}_{s,t}$ that are substantially different from $\hat{\mathbf{F}}_{i,j}$ will tend to be minimized by $\mathbf{A}(s, t)$ and will have MSEs that fall close to the upper bound of the histogram. Subimages that satisfy H^1 , if they exist, will be found in the tail of the distribution. If the variance of the noise is known, one can predict a range for this Gaussian, but in the absence of this knowledge it is necessary to rely on an iterative process to eliminate $\hat{\mathbf{G}}_{s,t}$ that do not form a Gaussian distribution.

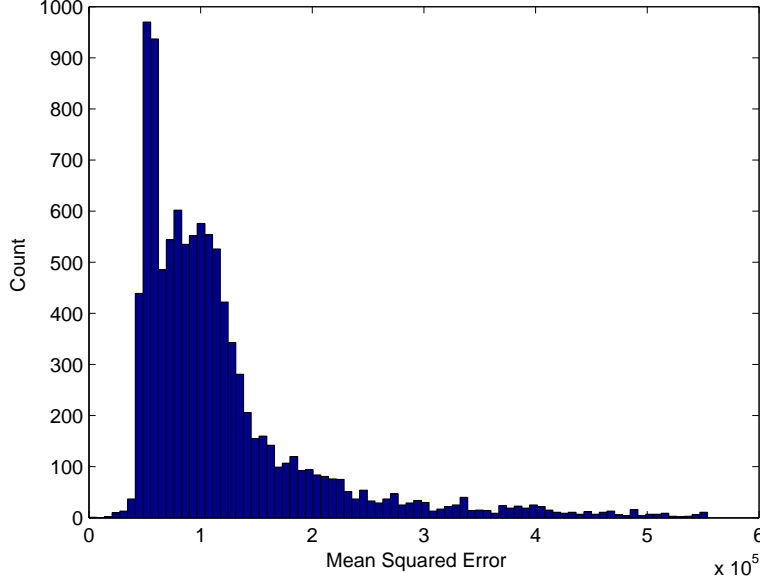


Figure 5.2: Histogram of the Mean Squared Error between a representative $\hat{\mathbf{F}}_{i,j}$ and all $\hat{\mathbf{G}}_{s,t}$. No scaling factor has been applied to $\hat{\mathbf{G}}_{s,t}$.

Beginning at the upper bound of the histogram of the MSEs, test the distribution of the MSEs that are less than γ using the Lilliefors test for normality, described in [18]. If the distribution is Gaussian, assign $\mathbf{A}(s, t) = 1$ for all $\hat{\mathbf{G}}_{s,t}$. If not, set $\{\mathbf{A}(s, t) = 0 \mid \gamma \leq \|\hat{\mathbf{F}}_{i,j} - \mathbf{A}(s, t)\hat{\mathbf{G}}_{s,t}\|_F^2 \leq \|\hat{\mathbf{F}}_{i,j}\|_F^2\}$ and $\{\mathbf{A}(s, t) = 1 \mid \|\hat{\mathbf{F}}_{i,j} - \mathbf{A}(s, t)\hat{\mathbf{G}}_{s,t}\|_F^2 < \gamma\}$ and reapply the Lilliefors test. Then, select γ based on the amount of processing time desired. This step is iterated for decreasing values of γ until either satisfied the Lilliefors test is satisfied for some number of $\hat{\mathbf{G}}_{s,t}$ or all $\hat{\mathbf{G}}_{s,t}$ have been eliminated as potential matches for $\hat{\mathbf{F}}_{i,j}$. In this way, for each $\hat{\mathbf{F}}_{i,j}$, a value for $\text{GDD}(\hat{\mathbf{F}}_{i,j}) = \langle \mathbf{A}(s, t)\hat{\mathbf{G}}_{s,t} \rangle$ is determined.

5.1.3.2 Image Restoration. To complete the algorithm for a single subimage, the block mean is restored by calculating

$$\hat{\mathbf{F}}_{i,j} = \text{GDD}(\hat{\mathbf{F}}_{i,j}) + \frac{1}{N^2} \sum_{u \in \mathcal{U}, v \in \mathcal{V}} \mathbf{F}_{i,j}(u, v).$$

It is then necessary to restore the image from $\hat{\mathbf{F}}_{i,j}$ over all (i, j) . One simple way to approach this is to use the restoration approach of the NLM and say

$$\hat{\mathbf{I}}(i, j) = \hat{\mathbf{F}}_{i,j}(n + 1, n + 1).$$

The problem with this approach in the GDD is that the original calculation of $\hat{\mathbf{F}}_{i,j}$ involved subtracting the block mean. As discussed in Appendix B, the sample mean for a block of zero mean noise is subject to some variation. Consequently, some portion of $\hat{\mathbf{F}}_{i,j}$ is still attributable to noise. Experimentally it was determined that it is advantageous to restore the image by summing and then averaging overlapping $\hat{\mathbf{F}}_{i,j}$ across the entire image.

It is also possible, and computationally advantageous, to obtain good results by denoising a subset of all possible $\mathbf{F}_{i,j}$ and combining these to form a single image. For example, in a 150×150 sized image, it would be possible to denoise 100 non-overlapping 15×15 blocks and recombine them to form a single image. The disadvantage to this approach is that the recombined images have undesirable discontinuities at the edges of the block due to an uneven restoration of the means. These discontinuities can be mitigated by choosing a latticed and overlapping set of $\mathbf{F}_{i,j}$ and averaging their denoised values together. This averaging makes the discontinuities between blocks less detectable and less objectionable as shown in Figure 5.10.

5.1.4 Experimental Results with the GDD Algorithm. The algorithm was used on simulated data and the results were compared with results obtained using the NLM algorithm. The data used for the test was a Light-Radar (LIDAR) image of a truck-mounted resolution board as a truth image (Figure 5.4) which was then corrupted with AWGN with $\sigma = 25$. The resulting noisy image is shown in Figure 5.5, and the histogram of the noisy image is shown in Figure 5.6. A 15×15 subimage was used for $\mathbf{F}_{i,j}$ to obtain the results shown in (Figure 5.7) for the NLM algorithm and (Figure 5.8) for the GDD method.

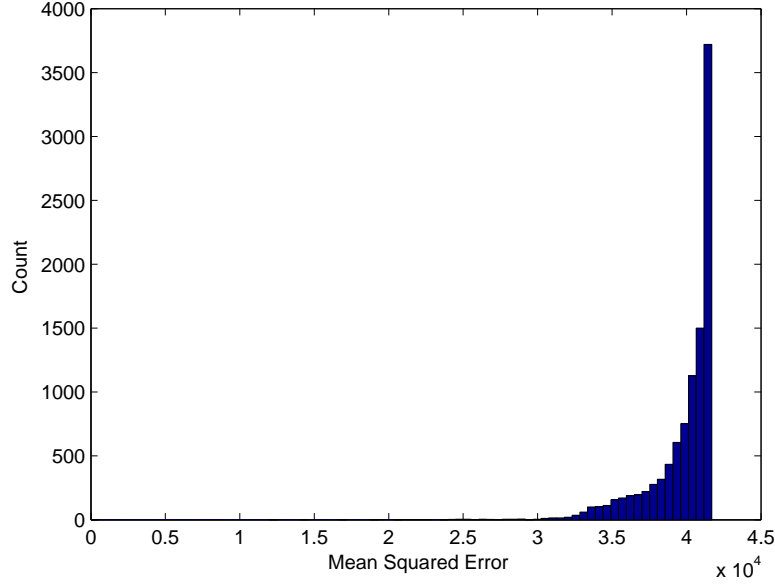


Figure 5.3: Histogram resulting from the calculation of $\left\| \hat{\mathbf{F}}_{i,j} - \mathbf{A}(s,t) \hat{\mathbf{G}}_{s,t} \right\|_F^2$ for a single $\hat{\mathbf{F}}_{i,j}$ and all values of $\mathbf{A}(s,t) \hat{\mathbf{G}}_{s,t}$. Note the difference from Figure 5.2 caused by applying the scaling factor $\mathbf{A}(s,t)$.

For the test image employed, the SNR provides a fair comparison of these two techniques. Annotating the truth image as \mathbf{I} , the mean of the truth image as $\bar{\mathbf{I}}$, the image being denoised as \mathbf{D} , and the output of the denoising algorithm as $\hat{\mathbf{I}}$, the SNR of the noisy image is calculated as

$$SNR = \frac{\left\| \mathbf{I} - \bar{\mathbf{I}} \right\|_F^2}{\left\| \mathbf{I} - \mathbf{D} \right\|_F^2}.$$

The SNR of the denoised image is calculated as

$$SNR = \frac{\left\| \mathbf{I} - \bar{\mathbf{I}} \right\|_F^2}{\left\| \mathbf{I} - \hat{\mathbf{I}} \right\|_F^2}.$$

The SNR of the original image after corruption with noise of $\sigma = 25$ was 20.92. Using the NLM algorithm, an SNR of 27.97 was achieved. Using the proposed method,

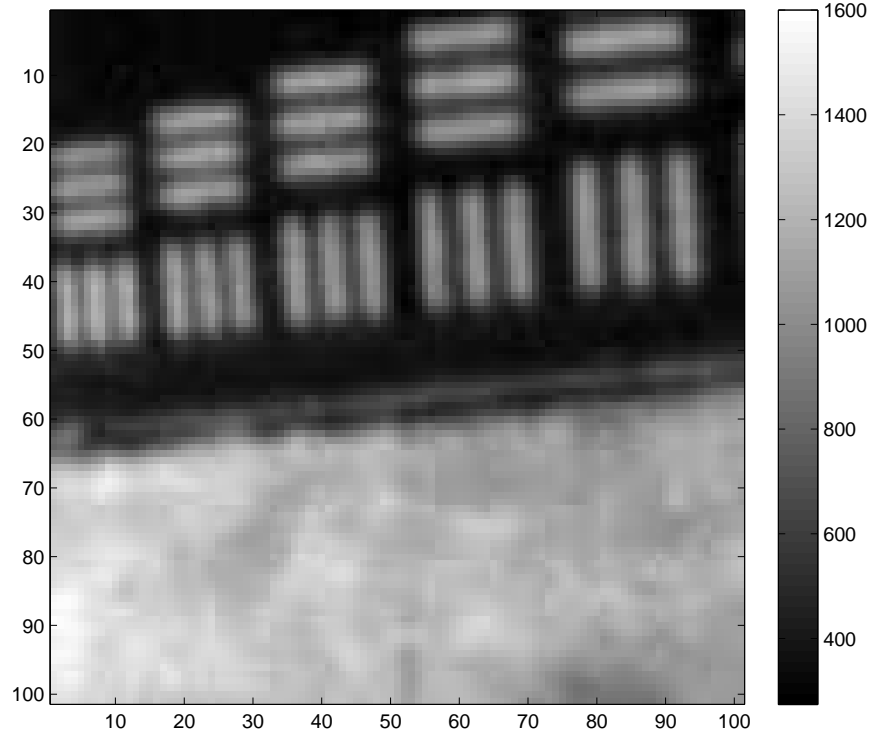


Figure 5.4: Truth image used to generate the simulation data. The bar to the right of the image indicates pixel values.

an SNR of 41.77 was achieved which represented a 49% improvement over the NLM algorithm.

Image denoising was also attempted using alternative strategies in an attempt to minimize the computational complexity by minimizing the number of $\mathbf{F}_{i,j}$ used in the algorithm. A first attempt was made to select $\mathbf{F}_{i,j}$ that were mutually exclusive. For an a 101×101 image with zero padding, it was possible to denoise the image using 49 versions of $\mathbf{F}_{i,j}$ and 10201 versions of $\mathbf{G}_{s,t}$. The results of this denoising are shown in Figure 5.9. Although the SNR in this case was 34.24, the image contained discontinuities resulting from the mean restoration process. In an attempt to minimize these discontinuities, an overlapping lattice of 98 $\mathbf{F}_{i,j}$ was used with 10201 $\mathbf{G}_{s,t}$. The results were then locally averaged together to form the image in in Figure 5.10 which has an SNR of 43.81.

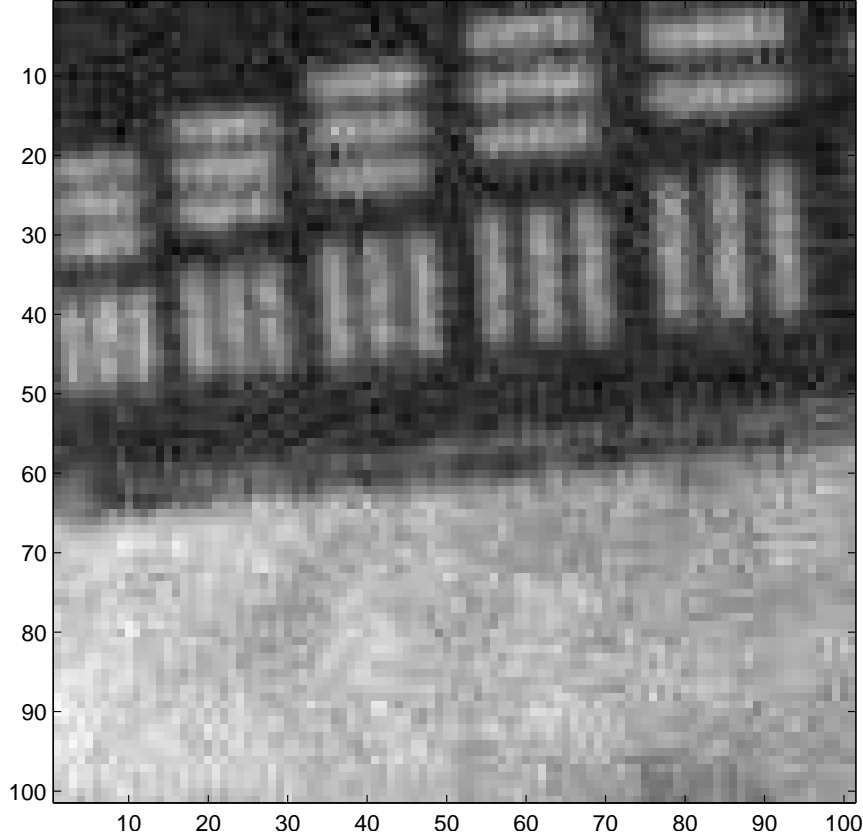


Figure 5.5: Truth image with AWGN of $\sigma = 25$ added.

5.1.5 Conclusions Drawn from Initial Results. This section introduced a unique image processing algorithm. Although it improved over results obtained using the NLM algorithm, it more importantly provided additional research opportunities. Notably, the chi-square distribution becomes more pronounced with fewer degrees of freedom and the probability of statistically similar blocks is expected to increase for smaller block sizes. By decreasing the neighborhood to some optimal size, it is expected that the performance of the algorithm can be improved. In addition, the results reflected here do not account for the averaging of permutations (e.g. rotations and translations) of $\mathbf{G}_{s,t}$ blocks which may provide an additional performance improvement.

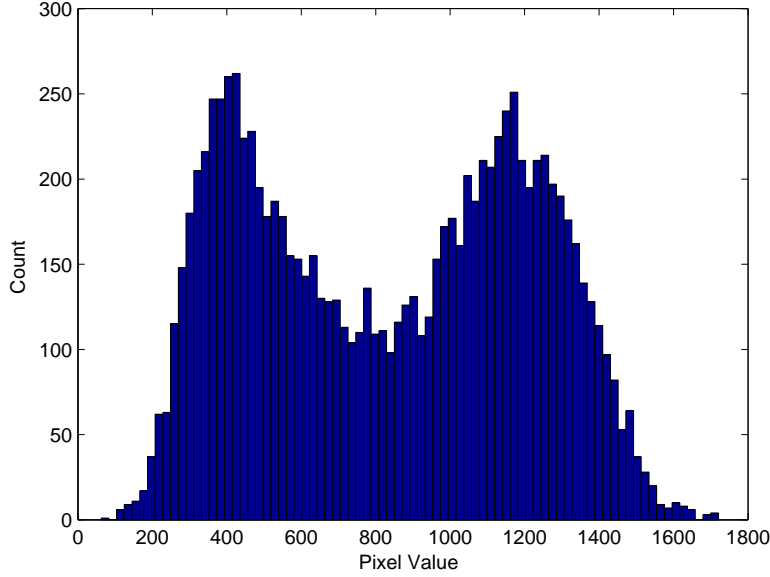


Figure 5.6: Histogram of the noisy image.

This implementation also assumed no *a priori* knowledge of the noise distribution; however, in many applications the distribution can be obtained through measurements of output from a given imaging system.

One significant problem with this method was that it was computationally expensive and processing times were significant for even small images. A second problem is that the results, while good, are less impressive than other state-of-the-art denoising methods. Consequently, other algorithms were developed.

5.2 The HOD and XCD Denoising Algorithms

This section describes two additional novel methods for denoising images. The algorithms operate by identifying regions of interest within a noise-corrupted image and then creating noise free estimates of the regions as averages of similar regions in the image. These similar regions are found by comparing examining the statistics of the error functions between the given region and other, identically sized regions in either the same image or in other images from the same sensor. The statistically

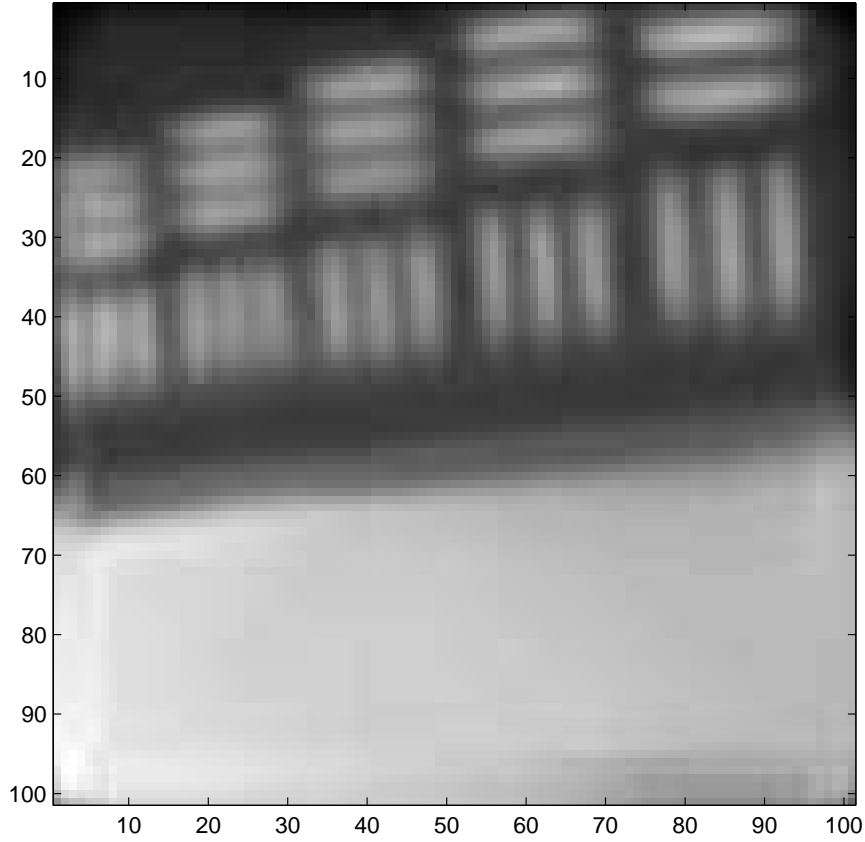


Figure 5.7: Output obtained using the NLM algorithm. SNR for this image was 27.97.

similar regions are averaged together to produce an estimate of the noise-free version of the region of interest. This technique is similar to multiframe averaging; however, only a single frame is required. The techniques are shown to outperform Wiener and median filtering over a wide range of noise conditions but are most effective in images with very low signal-to-noise ratios.

Section 5.2.1 describes a denoising method that denoises images using the first, second and third moments of regions within the image. Then, Section 5.2.2 describes a denoising method that uses the first and second moments of the data in concert with fast projection-based cross-correlations. Section 5.2.3 describes the algorithms and performance in comparison to Wiener filtering [26] and median filtering [26]

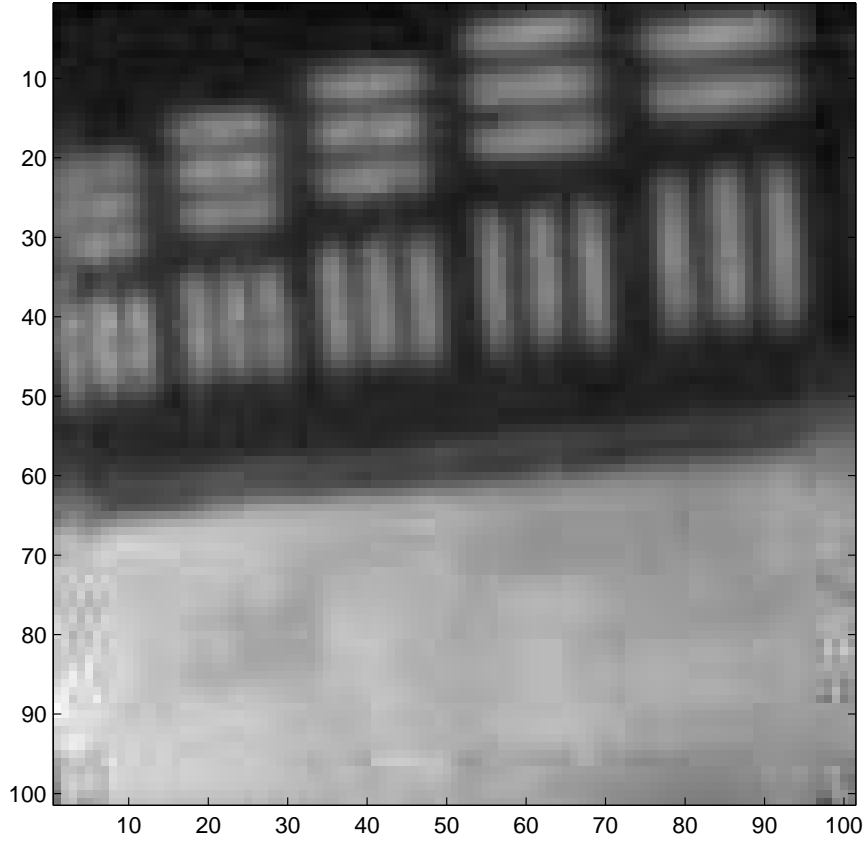


Figure 5.8: Output obtained using the GDD method. SNR for this image was 41.77.

when used on standard benchmark images as well as on actual LIDAR data with a simulated noise component.

5.2.1 Higher-Order Statistics Method for Block Matching. This section discusses using a method that employs the variances of the blocks and the skewness of their error functions as measures of block similarity. This algorithm is referred to as the Higher-Order Denoising (HOD) algorithm. This algorithm looks at the second moment of the block and pairs of blocks whose error function has a third-order moment indicative of a Gaussian distribution. A summary overview of the algorithm is provided followed by the details of the algorithm.

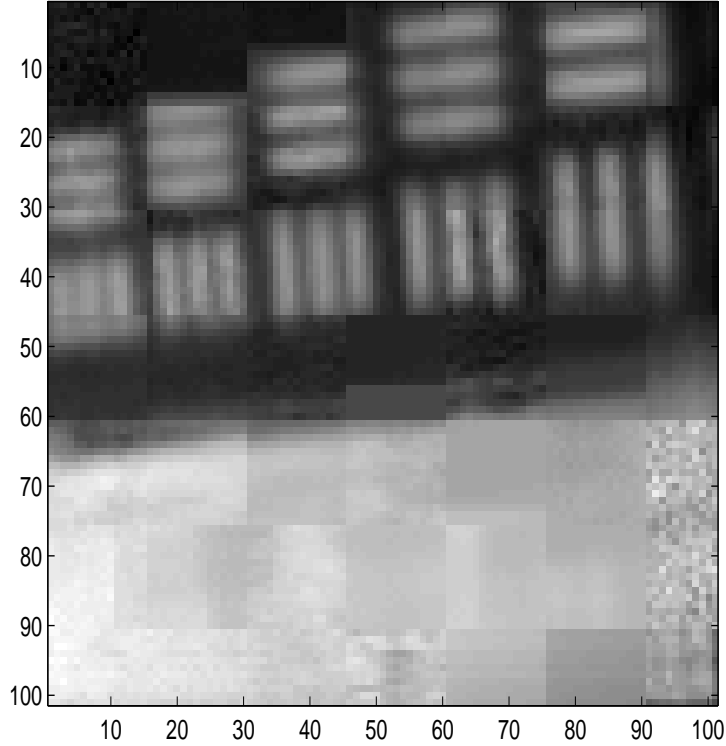


Figure 5.9: Output obtained by GDD denoising using 49 of 10201 possible 15×15 blocks. Note the block discontinuities that are similar to those obtained in JPEG restoration.

5.2.1.1 Overview of the Denoising Algorithm based on Higher Order Statistics.

1. Within an image \mathbf{D} , select an $N \times N$ -sized subimage indexed at the pixel (i, j) within the image to denoise. Call this block $\mathbf{F}_{i,j}$.
2. Develop estimates for the maximum variance of flat regions of the image, the maximum variance of the error between two featureless blocks, and the skewness of an ensemble of noise values.
3. For every pixel centered at (s, t) in the parent image, define an $N \times N$ neighborhood around the pixel and call this block $\mathbf{G}_{s,t}$.

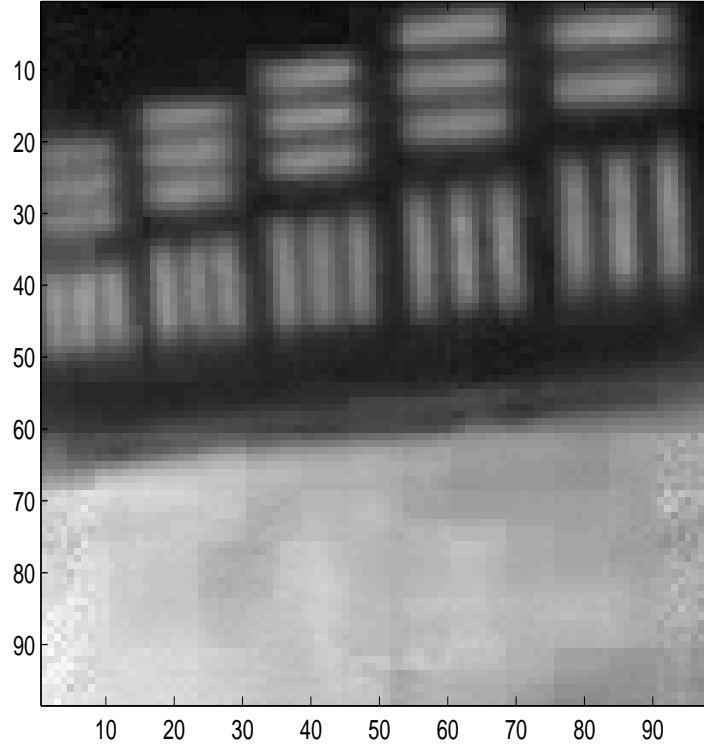


Figure 5.10: Output obtained by GDD denoising using an overlapping lattice of 98 of 10201 possible 15×15 blocks. Discontinuities still exist but are less objectionable than in Figure 5.9.

4. Subtract the means from all $\mathbf{F}_{i,j}$ and all $\mathbf{G}_{s,t}$.
5. For each block $\mathbf{F}_{i,j}$, evaluate whether the variance of the block is less than the upper limit of the variance of a block of noise only. If so, set the entire block to the block mean (i.e. zero).
6. If the variance of the block is above the threshold used in 5) above, calculate the error between the block and all other blocks in the image. If the variance and skewness of the error function between $\mathbf{F}_{i,j}$ and any $\mathbf{G}_{s,t}$ are within the allowable thresholds, include $\mathbf{G}_{s,t}$ in an average of blocks. Calculate the processed block as the average of $\mathbf{F}_{i,j}$ and all identified $\mathbf{G}_{s,t}$.

7. Add the mean of the original block to the processed block. This is the restored block.
8. Repeat steps 1-7 for all other blocks $\mathbf{F}_{i,j}$ in the image.
9. Reconstruct the image as an average of all the overlapping restored blocks across the image.

5.2.1.2 Description of the Higher Order Statistical Denoising (HOD)

Algorithm. In this subsection a more detailed overview of the image model, underlying assumptions, and the mathematical framework of the HOD algorithm is provided. The image model assumes that the predominate noise source is Additive White Gaussian Noise (AWGN) and that the noise is independent and identically distributed in each pixel. Define $\mathbf{I}_{i,j}$ as a diffraction-limited, $N \times N$ block of an image where $N \in \mathbb{N}$ and $N > 1$. The coordinate pair (i, j) indicates the location of center pixel of the neighborhood $\mathbf{I}_{i,j}$ within a larger image \mathbf{I} (i.e. $\mathbf{I}_{i,j} \subset \mathbf{I}$). If \mathbf{I} is corrupted by zero-mean Gaussian noise so that for each pixel, $\mathbf{I}_{i,j}(u, v)$ where $u \in \{1, \dots, N\}$ and $v \in \{1, \dots, N\}$, then

$$\mathbf{D}_{i,j}(u, v) = \mathbf{I}_{i,j}(u, v) + \mathbf{Q}_{i,j}(u, v),$$

where $\mathbf{D}_{i,j}$ is a subimage centered at (i, j) and $\mathbf{Q}_{i,j}$ is the realization of the Gaussian noise within that subimage. In matrix notation, this can be denoted as

$$\mathbf{D}_{i,j} = \mathbf{I}_{i,j} + \mathbf{Q}_{i,j}.$$

Within an image, there may be other $N \times N$ blocks centered at coordinates (s, t) that satisfy the equation

$$\mathbf{D}_{s,t} = \mathbf{I}_{i,j} + \mathbf{Q}_{s,t}.$$

If these subimages exist, then $\Delta_{s,t}$, the error between $\mathbf{D}_{i,j}$ and $\mathbf{D}_{s,t}$, can be calculated

$$\begin{aligned}
\Delta_{s,t} &= \mathbf{D}_{i,j} - \mathbf{D}_{s,t} \\
&= (\mathbf{I}_{i,j} + \mathbf{Q}_{i,j}) - (\mathbf{I}_{i,j} + \mathbf{Q}_{s,t}), \\
&= \mathbf{Q}_{i,j} - \mathbf{Q}_{s,t}.
\end{aligned} \tag{5.9}$$

For a single pixel in Δ ,

$$\Delta_{s,t}(u, v) = \mathbf{Q}_{i,j}(u, v) - \mathbf{Q}_{s,t}(u, v).$$

The noise characteristics of most individual sensors can be determined by empirical measurement and it is reasonable to assume that this information is available to the algorithm. In the algorithm proposed in this section, it is assumed that the variance of the predominant noise in the image *a priori* is known. Given the variance of the noise, σ , the statistical characteristics of Δ can be used to identify and average similar blocks in a given image. The values of the moments of \mathbf{Q} and Δ can be viewed as random variables and it is then possible to select matching blocks based on the values of these moments. For example, σ_Q^2 , the variance of an $N \times N$, zero mean, block of noise \mathbf{Q} can be calculated as:

$$\sigma_Q^2 = \langle \mathbf{Q}_n^2 \rangle, \tag{5.10}$$

where \mathbf{Q}_n^2 represents squares of the individual elements of an $N \times N$ block \mathbf{Q} . Assuming that the noise in the image is AWGN, this sum of terms is recognized to be a chi-square random variable.

Recalling that the limiting case of the chi-square distribution as N tends to infinity is a Gaussian distribution [28], the PDF of the variance of the measured noise in a block \mathbf{Q} may be approximated as a Gaussian with $\mu = \sigma^2$ and $\sigma_Q^2 = 2\sigma^4/N^2$. Using this approach, calculate three standard deviations from μ and roughly predict

the upper bound on the PDF of the variance of \mathbf{Q} as

$$\sigma_{max}^2 = \sigma^2 + 3\sqrt{\frac{2\sigma^4}{N^2}}. \quad (5.11)$$

A similar approach can be used to estimate the upper bound of the PDF of the variance of Δ which is denoted $\sigma_{\Delta_{max}}^2$. This bound can be calculated as

$$\sigma_{\Delta_{max}}^2 = 2\sigma_{\mathbf{Q}}^2 + 3\sqrt{\frac{8\sigma_{\mathbf{Q}}^4}{N^2}}. \quad (5.12)$$

More precise methods for determining the quartiles of a chi-square distribution are described in [28] but for this application more precise methods appear unnecessary for most block sizes.

Estimation of the skewness for a block of noise is more difficult as it involves the calculation of the third central moment of an ensemble of Gaussian random variables, however, a Monte Carlo simulation was employed to arrive at a polynomial function of block size to estimate skewness. Figure 5.11 shows the results of the Monte Carlo simulation and shows the measured maximum skewness for various block sizes and noise values. This maximum skewness is independent of the variance of the noise. The magnitude of the bound on the skewness of the error function is represented as \mathcal{S}_{max} , where the skewness of an ensemble of random variables \mathbf{Q} of size $n \times n$ and with mean $\mu_{\mathbf{Q}}$ is defined as

$$\mathcal{S}_{\mathbf{Q}} = \frac{N^4 \sum_{i=1}^n \sum_{j=1}^n (\mathbf{Q}(i, j) - \mu_{\mathbf{Q}})^3}{(\sum_{i=1}^n \sum_{j=1}^n (\mathbf{Q}(i, j) - \mu_{\mathbf{Q}})^2)^3}. \quad (5.13)$$

Once these limits are determined, processing of the image can begin. Define an image \mathbf{D} of size $S \times T$ where $S \in \mathbb{N}$ and $T \in \mathbb{N}$. Also, take an $N \times N$ -sized subimage $\mathbf{F}_{i,j} \subset \mathbf{D}$ where $\mathbf{F}_{i,j}$ is centered at $i \in \mathcal{S} \triangleq \{1, \dots, S\}$, $j \in \mathcal{T} \triangleq \{1, \dots, T\}$ and where $\{N = 2n + 1 \mid n \in \mathbb{N}\}$. If the image \mathbf{D} is zero padded by n in all directions, then for all $s \in \mathcal{S}$ and $t \in \mathcal{T}$, there are $N^2 - 1$ other subimages in \mathbf{D} which may

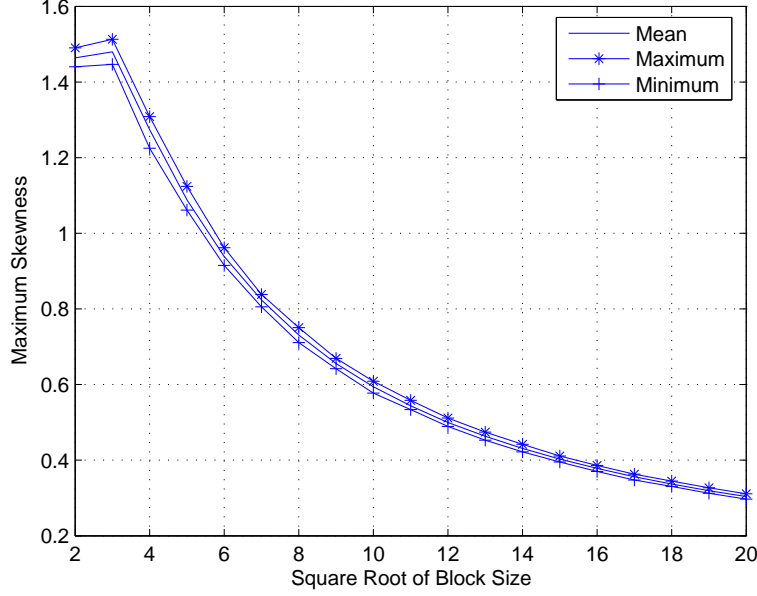


Figure 5.11: Plot of the measured maximum skewness vs. N - the square root of the block size. The plot displays the mean, maximum and minimum measured values of maximum skewness for various block sizes of AWGN with $\sigma_Q = 5$ to 150.

be similar in an L2-norm sense to \mathbf{F} and one subimage where $s = i$ and $t = j$. These subimages are denoted $\mathbf{G}_{s,t}$. Furthermore, call the index of each pixel in these subimages (u, v) where $u \in \mathcal{U} \triangleq \{1, \dots, N\}$ and $v \in \mathcal{V} \triangleq \{1, \dots, N\}$. In an attempt to remove illumination and reflectance differences, the means are removed from both the $\mathbf{F}_{i,j}$ and $\mathbf{G}_{s,t}$. For all x and y :

$$\hat{\mathbf{F}}_{i,j}(x, y) = \mathbf{F}_{i,j}(x, y) - \frac{1}{N^2} \sum_{u \in \mathcal{U}, v \in \mathcal{V}} \mathbf{F}_{i,j}(u, v) \quad (5.14)$$

and

$$\hat{\mathbf{G}}_{s,t}(x, y) = \mathbf{G}_{s,t}(x, y) - \frac{1}{N^2} \sum_{u \in \mathcal{U}, v \in \mathcal{V}} \mathbf{G}_{s,t}(u, v) \quad \forall s \in \mathcal{S}, t \in \mathcal{T}. \quad (5.15)$$

Most natural images have significant low-frequency content that can be de-noised using first-order statistics. Where the measured variance of a block is less

than the maximum expected variance of a block of noise, a denoised version of the block is estimated by replacing all the pixels of the block with the block mean. Specifically, for any block $\mathbf{F}_{i,j}$ that has $\sigma_{\mathbf{F}_{i,j}}^2 < \sigma_{max}^2$, a noiseless version of each pixel in $\mathbf{F}_{i,j}$ is estimated as

$$\text{HOD}(\mathbf{F}_{i,j})(u, v) = \frac{1}{N^2} \sum_{u \in \mathcal{U}, v \in \mathcal{V}} \mathbf{F}_{i,j}(u, v). \quad (5.16)$$

Regions that have significant high frequency content are denoised by identifying and averaging a subset of blocks $\mathbf{G}_{s,t}$ that have similar statistical characteristics. This set of blocks with similar characteristics is represented as B and the s, t pair corresponding to a member of B is as β . A denoised version of block $\mathbf{F}_{i,j}$ is then constructed as

$$\text{HOD}(\mathbf{F}_{i,j}) = \frac{1}{|B| + 1} \left(\hat{\mathbf{F}}_{i,j} + \sum_{\beta \in B} \hat{\mathbf{G}}_{s,t} \right) + \frac{1}{N^2} \sum_{u \in \mathcal{U}, v \in \mathcal{V}} \mathbf{F}_{i,j}(u, v), \quad (5.17)$$

where the last term is the mean that was previously subtracted in (5.14). Given the error vector between two blocks $\Delta_{s,t}$ with mean $\mu_{\Delta_{s,t}}$, the members of B are those $\hat{\mathbf{G}}_{s,t}$ where

$$\frac{1}{N^2} \sum_{u \in \mathcal{U}, v \in \mathcal{V}} (\Delta_{s,t}(u, v) - \mu_{\Delta_{s,t}})^2 < \sigma_{\Delta_{max}}^2, \quad (5.18)$$

and where the skewness of the error function is evaluated to determine if

$$\left| \frac{N^4 \sum_{u \in \mathcal{U}, v \in \mathcal{V}} (\Delta_{s,t}(u, v) - \mu_{\Delta_{s,t}})^3}{(\sum_{u \in \mathcal{U}, v \in \mathcal{V}} (\Delta_{s,t}(u, v) - \mu_{\Delta_{s,t}})^2)^3} \right| < |\mathcal{S}_{max}|. \quad (5.19)$$

Using this approach, all $\hat{\mathbf{F}}_{i,j}$ in an image are evaluated against all $\hat{\mathbf{G}}_{s,t}$. An estimate of the noise-free image must now be created from the noise free estimates of the individual blocks in the image. The algorithm concludes by reconstructing the image as the average of all overlapping blocks $\text{HOD}(\mathbf{F}_{i,j})$.

5.2.2 Correlation-Based Method for Block Matching. This section discusses using a method that employs the variances of the blocks and their error functions in conjunction with a method for evaluating the similarity of the blocks based on their cross correlation peaks. This method is referred to as the cross correlation denoising (XCD) algorithm. The algorithm looks at the second moment of the block and evaluates whether comparative blocks are spatially correlated with the block under study. In an effort to reduce processing time, the algorithm replaces the calculation of the third moment with a projection-based correlation to determine whether or not the peak correlation of two blocks is located at the center of their cross correlation.

5.2.2.1 Overview of the Cross Correlation Denoising Algorithm.

Most steps of the XCD algorithm are identical to those in the HOD algorithm described in section 5.2.1; however, instead of calculating the skewness of the error vector between two blocks in step 6, the algorithm uses the projections of the blocks to calculate their cross correlations and observe the location of the cross correlation peak. The processing steps of the XCD algorithm are:

1. Within an image \mathbf{D} , select an $N \times N$ -sized subimage indexed at (i, j) within the image to denoise. Call this block $\mathbf{F}_{i,j}$.
2. Develop estimates for the maximum variance of flat regions of the image and the maximum variance of the error between two featureless blocks.
3. For every pixel centered at (s, t) in the parent image, define an $N \times N$ neighborhood around the pixel and call this block $\mathbf{G}_{s,t}$.
4. Subtract the means from all the blocks.
5. For each block, evaluate whether the variance of the block is less than the upper limit of the variance of a block of noise only. If so, set the entire block to the block mean.

6. If the variance of the block is above the threshold used above, calculate the error between the block and all other blocks in the image. If the variance of the error function between $\mathbf{F}_{i,j}$ and any $\mathbf{G}_{s,t}$ are within the allowable thresholds, calculate the cross correlations of the projections. If the cross-correlation peaks of the projections are in their centers, include $\mathbf{G}_{s,t}$ in an average of blocks. Calculate the processed block as the average of $\mathbf{F}_{i,j}$ and all identified $\mathbf{G}_{s,t}$.
7. Add the mean of the original block to the processed block. This is the restored block.
8. Repeat step 1-7 for all other blocks in the image.
9. Reconstruct the image as an average of all the restored blocks in the image.

5.2.2.2 Description of the Correlation-Based Denoising (XCD) Method for Block Matching.

An alternative method of block selection, which is also based on L2 distance, is also effective. In many cases, blocks within an image are most similar, in an L2 sense, to shifted versions of themselves. Although these shifted blocks are close in L2 distance, they may introduce structurally different blocks into a block average thereby biasing the result. This is especially true along edges of image features. When the shift is in the direction of an edge, this contributes constructively to a block averaging algorithm. When the shift is perpendicular to an edge but the resulting block is close in L2 distance, it has the effect of smoothing the edges in a denoised block and thereby reducing high-frequency image content.

Blocks that are close in L2 distance may be predicted by looking at the autocorrelation of a block being denoised. The primary peak of the autocorrelation corresponds to the $[0, 0]$ shift. The subpeaks with magnitudes less than the primary peak correspond to the center pixels of blocks whose that are shifted versions of a block of interest and are close that block in L2 distance. This observation suggests that it may be possible to find blocks that have similar content by considering the location of the peak of the cross correlation of two blocks that are close in L2 dis-

tance. This may be a computationally expensive approach; however, good results can be attained while minimizing computational load by using 1-D projection-based cross-correlations instead of 2-D cross-correlations.

For image registration applications, Cain *et al.* [11] described an algorithm that uses the x and y projections of an image to find their 2-D cross correlation peak. In this algorithm, the cross-correlation peaks of the one-dimensional projections of two images corresponds to the x and y coordinates of the 2-D cross correlation. This is shown to be a computationally-efficient alternative to the more traditional approach of finding the two-dimensional cross-correlation peak of the images. In order to exploit the computational efficiency of this approach in the XCD algorithm, the x and y projections of each block are calculated as a preprocessing step when the blocks are created. Then, for each $\mathbf{F}_{i,j}, \mathbf{G}_{s,t}$ pair whose measured variance is below the calculated maximum allowable variance, it is necessary to calculate two, 1-D cross correlations. For a given $\mathbf{G}_{s,t}$, if the cross correlation peak is centered in both 1-D cross correlations, $\mathbf{G}_{s,t}$ is included in the average of similar blocks.

Another set of blocks with $\langle \Delta_{s,t}^2 \rangle < \sigma_{\Delta_{max}}^2$ can now be constructed that have centered cross-correlation peaks for both row and column projections. This set is denoted Γ and the s, t pair corresponding to a member of Γ is γ . Using the set Γ , a denoised version of block $\mathbf{F}_{i,j}$ can be constructed as

$$XCD(\mathbf{F}_{i,j}) = \frac{1}{|\Gamma| + 1} \left(\hat{\mathbf{F}}_{i,j} + \sum_{\gamma \in \Gamma} \hat{\mathbf{G}}_{s,t} \right) + \frac{1}{N^2} \sum_{u \in \mathcal{U}, v \in \mathcal{V}} \mathbf{F}_{i,j}(u, v). \quad (5.20)$$

As in the previous section, the restored image is constructed as an average of all the denoised estimates of the blocks.

5.2.3 HOD and XCD Simulation Results. This section presents results using LIDAR and standard benchmark images as truth with additive Gaussian noise. These results demonstrate that the HOD and XCD denoising methods can suc-

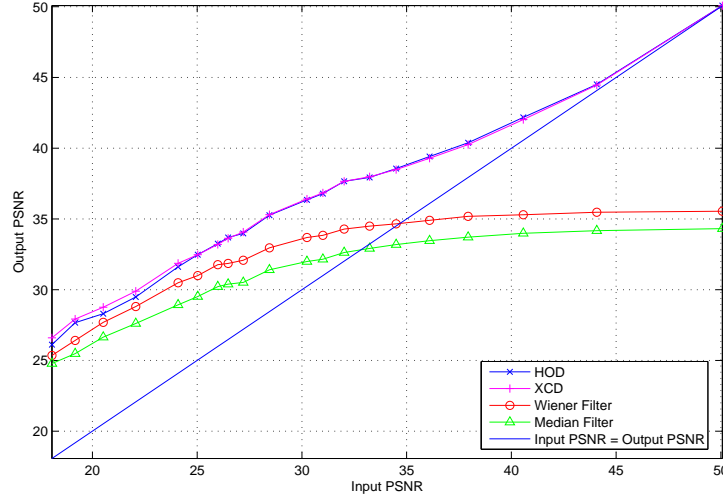


Figure 5.12: Graph of results comparing output of HOD filtering, XCD filtering, Wiener filtering and median filtering of the 101×101 image shown in Figure 5.5 over a wide range of noise values. Typical block sizes (6×6) are chosen for the HOD and XCD algorithm. The line in the graph labelled “Input PSNR = Output PSNR” shows the point where denoising methods produce results with lower PSNR than the original image. The HOD and XCD methods approach but do not reach this line.

cessfully denoise images with results that are consistently favorable to Wiener and median filtering and on par with many wavelet denoising methods. In the following discussion and graphs, results are presented using PSNR where the PSNR of an $S \times T$ sized image is defined as:

$$PSNR = 10 \log_{10} \left(\frac{(I_{max}^2)}{\frac{1}{ST} \|\mathbf{I} - \hat{\mathbf{I}}\|_F^2} \right), \quad (5.21)$$

where \mathbf{I} is the diffraction-limited image, $\hat{\mathbf{I}}$ is an estimate of the image, I_{max} is the maximum value found in the image \mathbf{I} .

The performance of the algorithm was examined using various images. The variable parameters in these simulations were the individual images, the variance of the AWGN and the block sizes used in the denoising algorithms. Results are presented for each of the two algorithms.

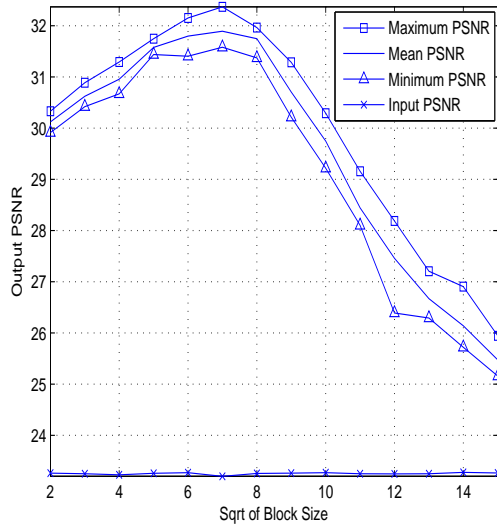


Figure 5.13: Block size vs. HOD output for a 101×101 image. For each block size, the image was corrupted with ten noise realizations of $\sigma_{noise} = 110$ (mean input PSNR = 23.26) and denoised. The maximum, minimum and mean PSNRs for 10 runs using each block size are plotted.

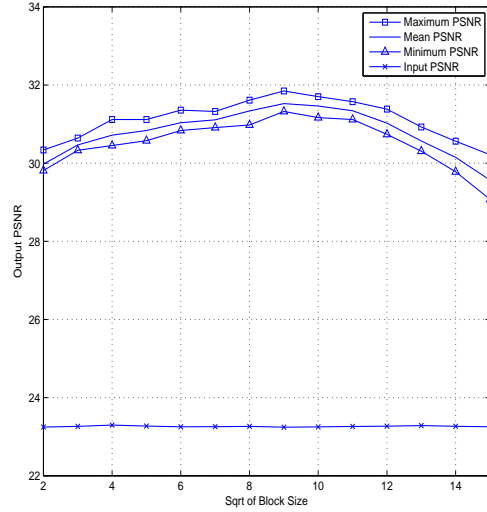


Figure 5.14: Block size vs. XCD output for the 101×101 image. As in 5.13, the image was corrupted with ten noise realizations of $\sigma_{noise} = 110$ (mean input PSNR = 23.26) and denoised. The maximum, minimum and mean PSNRs for 10 runs using each block size are plotted.

Experimentation began with the LIDAR image shown in Figure 5.5 and examined the effectiveness of the algorithm with various block sizes and noise values. The results of the algorithms across a range of input noise values are shown in Figure 5.12. This graph shows the results in comparison to optimally-sized Wiener and median filters. The diagonal lines across the graph indicates the point where the output PSNR is equal to the input PSNR. To the right of this line, output PSNRs are less than input PSNRs indicating that Wiener filtering and median filtering actually degrade the image. On the left hand side of the diagonal line lies the region where all three algorithms improve the PSNR of an input image. The HOD and XCD algorithms outperform the optimally-sized Wiener and median filters (as implemented by MATLAB 7.1) across this region.

Output performance is also dependent upon block size. In (5.17) and (5.20), for any given block $\mathbf{F}_{i,j}$ the results of the algorithm are dependent upon the number of blocks $\mathbf{G}_{s,t}$ that contribute to the average in $HOD(\mathbf{F}_{i,j})$ and $XCD(\mathbf{F}_{i,j})$. Recalling that the data model is $\mathbf{D}_{i,j} = \mathbf{I}_{i,j} + \mathbf{Q}_{i,j}$, an average of blocks with identical \mathbf{I} but different \mathbf{Q} is expected to converge to \mathbf{I} as the number of blocks increases. In general, at small block sizes, it is more likely to find blocks $\mathbf{G}_{s,t}$ that are similar in underlying content to $\mathbf{F}_{i,j}$; however, a small sample size is not as likely to have higher-order statistics that are predicted by the model. As the block size increases, confident in the statistics increases, but it becomes less probable that blocks can be found with matching image content.

The graphs in Figure 5.13 and Figure 5.14 demonstrate the trade-off involved in choosing block size for the image shown in Figure 5.5 corrupted by AWGN. The graphs show the maximum, minimum and mean values for the output of ten different realizations of noise across various block sizes. For this image the optimal block size for the HOD algorithm is 7×7 and for the XCD algorithm is 9×9 . In general, a block size of approximately 6×6 provides good results. Figures 5.15 through 5.23 show output results using a 512×512 image created using frame averaging



Figure 5.15: 512×512 image of a tank derived from LIDAR data.



Figure 5.16: Image of Figure 5.15 corrupted with noise of $\sigma = 9000$, input PSNR = 18.96.



Figure 5.17: Image in Figure 5.16 denoised using HOD and a block size of six. Output PSNR = 33.17.



Figure 5.18: Image of Figure 5.17 denoised using Wiener filtering. Output PSNR = 26.23.

of LIDAR data. This image was corrupted with Gaussian noise and then denoised using HOD and Wiener filtering. Results for a 256×256 subregion of the image are shown in Figures 5.2.3 through 5.2.3. Beginning with the corrupted image shown in Figure 5.2.0, results for the Wiener filter shown in Figure 5.2.3 and for the HOD algorithm in Figure 5.2.3. The the results of the HOD algorithm are not only better in PSNR but are visually more appealing than median or Wiener filtering. Figures 5.15 through 5.23 show the results of filtering the entire 512×512 tank image with AWGN of $\sigma = 9000$ and input PSNR = 18.96. In this image, the advantage of HOD over Wiener filtering are even more apparent. For benchmarking purposes, the algorithms were also applied against the standard images *Lena*, *Barbara*, *Boats*, *House*, *Peppers* and compared with other denoising algorithms in the literature. The results of the HOD and XCD algorithms using a constant block size of 6×6 are shown in Tables 5.1 and 5.2.

The exemplar-based denoising algorithm described Kervrann and Boulanger [31, 32] is of interest because it is a non-transform domain algorithm that also uses L2 distance for block selection. Comparative results for the *Peppers* image are shown in Figure 5.24. Exemplar-based denoising performed better than both the HOD and XCD algorithms at relatively low noise levels ($\sigma < 70$) but did not perform as well at the higher noise levels that are common in passive infrared and LIDAR imaging.

The algorithms also compared favorably to the SUREshrink and Bayeshrink wavelet coefficient shrinkage algorithms that are described in [19] and [14] and are evaluated by Chang in [14] within a range of PSNRs from approximately of 17 to 28. The SUREshrink and Bayeshrink algorithms determine and apply thresholds to coefficients in the wavelet domain.

Overall, the XCD and HOD algorithms compared favorably with most denoising algorithms but fell short of the reported results for the most recent wavelet coefficient shrinkage algorithms that examine and combine neighborhood statistics including [15], [17], [42], and [44]. However, the algorithms proposed do provide a

σ / <i>Input PSNR</i>	Lena <i>512x512</i>	Barbara <i>512x512</i>	Boats <i>512x512</i>	House <i>256x256</i>	Peppers <i>256x256</i>
10/28.14	34.39	33.10	32.25	35.05	33.71
20/22.10	30.63	27.86	28.57	30.88	29.23
30/18.57	28.67	24.99	26.39	28.63	26.50
50/14.16	26.77	22.93	24.53	25.72	23.75
75/10.64	25.25	22.19	23.34	24.47	22.13
100/8.14	24.07	21.55	22.53	23.45	21.28
125/6.20	22.93	20.93	21.71	22.43	20.54
150/4.59	21.99	20.29	20.96	21.60	20.05

Table 5.1: Output PSNRs of the HOD method using a block size of six applied across several standard images with additive noise of varying standard deviation.

σ / <i>Input PSNR</i>	Lena <i>512x512</i>	Barbara <i>512x512</i>	Boats <i>512x512</i>	House <i>256x256</i>	Peppers <i>256x256</i>
10/28.14	34.45	32.98	32.46	35.05	33.72
20/22.10	30.77	27.70	28.72	30.98	29.42
30/18.57	28.91	24.94	26.75	28.87	26.54
50/14.16	26.93	23.01	24.80	26.13	24.26
75/10.64	25.30	22.20	23.46	24.61	22.50
100/8.14	24.06	21.54	22.56	23.44	21.36
125/6.20	22.89	20.90	21.70	22.37	20.56
150/4.59	21.94	20.26	20.94	21.55	20.05

Table 5.2: Output PSNRs of the XCD method using a block size of six applied across several standard images with additive noise of varying standard deviation.

mechanism for implementing neighborhood-based denoising in a manner that yields impressive results and could be relatively straightforward to implement in combinational logic.

5.3 Chapter Summary

This chapter of the dissertation has provided a review of recent image processing literature on single-frame image denoising and developed and demonstrated three similar block-based denoising algorithms. These algorithms exploited different measures of similarity between blocks than those used by other denoising algorithms in the literature. The algorithms described in this chapter are also different from

	Lena	Barbara	Boats	House	Peppers
<i>Method</i>	<i>512x512</i>	<i>512x512</i>	<i>512x512</i>	<i>256x256</i>	<i>256x256</i>
HOD	24.07	21.55	22.53	23.45	21.28
XCD	24.06	21.54	22.56	23.44	21.36
Exemplar [31]	23.32	20.64	21.78	23.08	20.51
Median Filter	15.77	15.33	15.68	15.63	15.68
Wiener Filter	15.46	15.24	15.43	15.41	15.39

Table 5.3: Results of various methods with input PSNR = 8.14 using a block size of six applied across several standard images with additive noise of varying standard deviation.

those described in the literature because they suspend the requirement for information included in the average to be located in close spatial proximity to the pixel being denoised. In all of the algorithms reviewed in Section 2.5, and in the more basic low-pass and median filters described in the introduction, close spatial proximity to a given pixel was an primary consideration in selecting other pixels to include in an average. The algorithms have been shown to achieve better results than many neighborhood filters by suspending this requirement. Computational load was also reduced in a number of areas by using binary weighting schemes.

Overall, the algorithms worked best and had the lowest processing times when dealing with images with significant amounts of noise (e.g. Input PSNR < 14). With less noise in an image, it becomes increasingly difficult to find blocks that are close in L2 distance with statistics that meet algorithm criteria. This observation may indicate a fundamental limit on denoising methods that rely on image statistics.

One of the more interesting aspects of both the methods introduced here and the techniques reviewed in Section 2.5 is the that the smoothing algorithms all generally produce results that exceed the CRLB. It is a relatively simple exercise to show that the maximum-likelihood estimate of an image \mathbf{I} which corrupted with AWGN with variance σ^2 is \mathbf{D} . It is also fairly simple to show that the variance on an estimate of a pixel in the image is also σ^2 . Block-based or other smoothing algorithms provide estimates that are, for most images, much better than a straightforward

CRLB calculation would predict by introducing additional assumptions about the frequency content of natural images. In particular, the underlying assumption is that most natural images contain predominantly low-frequency spatial content and that by averaging using linear or nonlinear filters a reasonable estimate of the images under study may be provided.

In addition to the methods noted in Section 2.5, many state-of-the-art wavelet coefficient shrinkage denoising methods, including those discussed in [17], [44], [42] among others, rely on combining of neighborhood information in the wavelet transform domain. Regardless of the domain used, in the presence of noise, the ability to combine information from these neighborhoods is subject to some degradation. Follow-on work may include investigation into the fundamental performance limits encountered by these algorithms.



Figure 5.19: Original 256×256 LIDAR image of a tank resulting from a multiframe average.

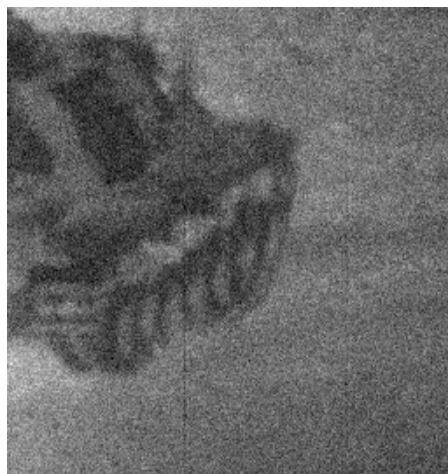


Figure 5.20: Tank image with Additive White Gaussian Noise, $\sigma_{noise} = 5000$, input PSNR = 19.33.

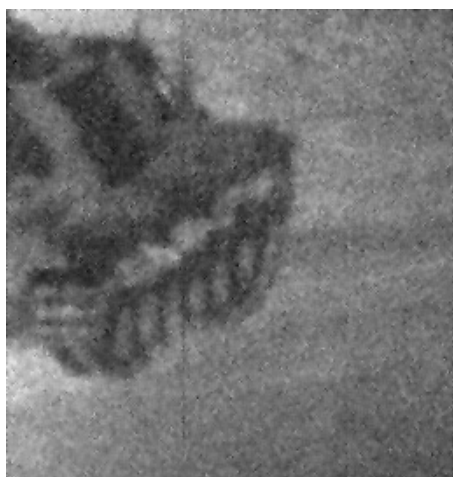


Figure 5.21: Tank image with Additive White Gaussian Noise, (input PSNR = 19.33) after Wiener filtering. Output PSNR = 26.43.



Figure 5.22: Tank image with Additive White Gaussian Noise, (input PSNR = 19.33) after HOD filtering with block size of 5, output PSNR = 31.00. Note that in addition to reducing stationary AWGN, non-stationary readout noise has also been reduced.

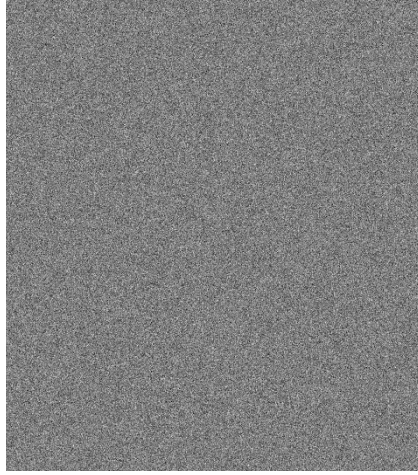


Figure 5.23: The “method noise” derived by subtracting the denoised image found in Figure 5.17 from the original noisy image found in Figure 5.16. Note the absence of feature content in this image. The actual value of σ was 8976. The measured value of σ in this method noise is 8801.

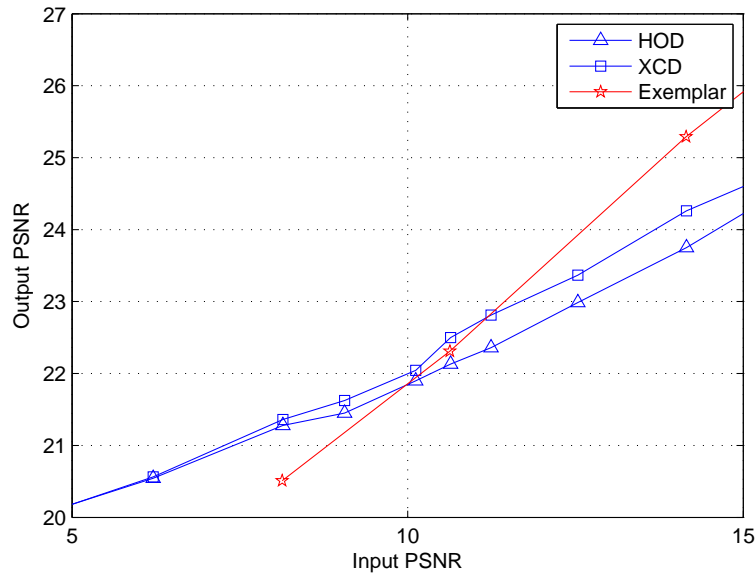


Figure 5.24: Results of the HOD and XCD methods compared against the Exemplar-based image denoising algorithm described in [31] using results reported in [31].

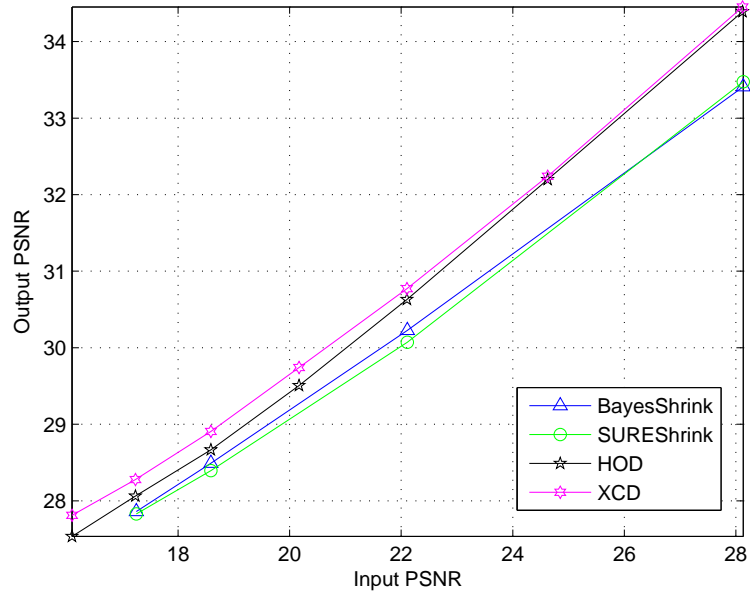


Figure 5.25: Results of the HOD and XCD methods compared against the Bayesshrink and SUREshrink algorithms described in [14] and [19] using results reported in [14].

VI. Conclusions

This dissertation has introduced research into image registration and single-frame denoising which has yielded novel image-processing algorithms and improved general theoretical understanding of bounds on the performance of shift estimators. The dissertation introduced methods to improve image quality and explored the theoretical limits of an algorithm's ability to achieve these improvements. In this final chapter, the results from the previous chapters are summarized and additional research is proposed that can extend the efforts described in the earlier chapters of this dissertation.

This chapter is organized as follows: In Section 6.1 a summary of the significant contributions of Chapters III, IV and V is provided. Then, in Section 6.2, areas that are believed to yield fruitful research that will extend the work performed in this dissertation are discussed.

6.1 Summary of Results and Contributions

This section provides an overview of contributions from the dissertation.

6.1.1 Review of Results in Chapter III. Chapter III provided a method for improving the performance of projection-based image registration algorithms at minimal computational cost. It explained how a low-pass filtering can be designed to exploit spatial correlations in an image and improve the performance of image registration algorithms. It also described experiments conducted with actual test data that have confirmed our analytical results.

The major contributions of this chapter included a generalization and modification of the FOM of Cain *et al.* [11] so that it could be applied to image projections containing correlated noise. This was necessary to apply the FOM to filtered projec-

tions the filtered case. This new FOM was then used in a procedure for finding the length of a boxcar filter which was applied to the projections of an image.

A second contribution of this chapter was the use of the FOM in a procedure that optimized a boxcar filter to minimize the mean-squared error of shift estimates created from projections of an image. The chapter showed that when uncorrelated noise is added to an image, the covariance structure of a projected image remains largely intact. A procedure for estimating the covariance function from available noise-corrupted data was introduced. The covariance function was then used in conjunction with the revised FOM to find the optimal length of boxcar filter that minimized the mean-squared error of shift estimates even in low-SNR environments.

The filters were compared with other low-pass filters and shown to be both computationally efficient and effective. Results showed an improvement by factors up to 5.5 in mean-squared error and a reduction by at least $\mathcal{O}(N)$ in computational complexity from 2-D methods. Further computational advantages were also discussed for this filtering method compared with other filtering methods described in the literature for reducing the mean-squared error of shift estimates.

6.1.2 Review of Results in Chapter IV. Chapter IV presented a calculation and comparison of theoretical performance bounds for image registration algorithms. It showed that for large images under conditions of full-frame registration, the CRLB is an adequate measure of performance for most realistic imaging conditions. For projected, optically filtered, or small images, however, the CRLB may not sufficiently predict bounds on the performance of a registration algorithm and the Barankin bound was introduced as a method for providing a more accurate estimate.

Chapter IV first examined the one-dimensional case of filtered projections and derived analytical expressions for the CRLB and Barankin bound of a shift estimate for two filtered and projected images of the same scene. This was compared with the CRLB and Barankin bound of shift estimates generated from 2-D data. The results

for the 1-D case were shown to be have much higher lower bounds on mean-squared error and, that for low-SNR projected images, there was a significant difference between the mean-squared error predicted by the CRLB and that predicted by the Barankin bound.

The chapter then examined and compared the effect of focal-length errors on the lower bounds of the mean-squared error of shift estimators. For the test images, the CRLB and Barankin bounds increased as the severity of focal-length errors increased in the simulations. In defocused imagery, the Barankin bound provided a higher estimate of SNR than that predicted by the CRLB even at moderate noise levels.

It is also worth noting that Cain *et al.* [11] show that shift estimates for low intensity images in the presence of fixed pattern noise may actually be better using projections than full 2-D estimates. Bounds on this behavior would also make an interesting future study.

Perhaps the most interesting aspect of the research documented in this paper was that the bounds under study were most applicable to distortions of small images. In many target-recognition applications, objects being imaged may be rotated and scaled and salient features may be extracting using various filtering techniques. This observation suggests many follow-on applications which are discussed next in Section 6.2.

6.1.3 Review of Results in Chapter V. In Chapter V, three similar block-based denoising algorithms were developed and demonstrated. These algorithms identified similar regions within a single image that could be used to create block averages in a way that was similar to the multiframe averaging facilitated by image registration in previous chapters. The new algorithms exploit different measures of similarity between blocks than those used by other denoising algorithms in the literature and use efficient binary weighting schemes in their block averages. These

algorithms are also different from those described in the literature because they suspend the requirement for information included in the average to be located in close spatial proximity to the pixel being denoised. The chapter showed that in low-SNR situations these algorithms could achieve better results than many neighborhood filters by suspending this requirement. Computational load was also reduced in a number of areas by using binary weighting schemes.

The three algorithms introduced in Chapter V were the Gaussian-Detection Denoising (GDD) algorithm, the Higher-Order Denoising (HOD) algorithm and the Cross-Correlation Denoising (XCD) algorithm. The GDD algorithm attempted to identify similar blocks by evaluating whether or not their error functions belonged to a Gaussian distribution. This algorithm did remove some noise in the image but was less successful than other methods, including the Wiener filter. This was not the case with the HOD algorithm. This algorithm looks at the second moment of the block and pairs of blocks whose error function has a third-order moment indicative of a Gaussian distribution. This algorithm consistently outperformed low-pass filtering techniques including the Wiener filter and was shown to be on-par with, and in some cases better than, other noise reduction algorithms found in current literature. The XCD algorithm was the third denoising algorithm developed and demonstrated in this chapter. This algorithm, attempted and succeeded in achieving the performance of the HOD algorithm, while using block projections to reduce the processing requirements of the computations. Most of the steps of the XCD algorithm were identical to those in the HOD algorithm; however, instead of calculating the skewness of the error vector between two blocks, the projections of the blocks were used to calculate their cross correlations and observe the location of the cross-correlation peak.

Overall, the algorithms worked best and had the lowest processing times when dealing with images with significant amounts of noise (e.g. input PSNR < 14 dB). With less noise in an image, it becomes increasingly difficult to find blocks that are

close in L2 distance with statistics that meet our criteria. This observation may indicate a fundamental limit on denoising methods that rely on image statistics. Potential follow-on work to the work described in this chapter is discussed in depth in Section 6.2 but may include investigation into the fundamental performance limits encountered by these algorithms.

6.2 *Recommended Future Research*

This section outlines additional research efforts that could be taken to extend the work described in this dissertation. Further research is described that could be performed in the areas of image registration, bounds on registration performance and block-based denoising.

6.2.1 Image Registration. Chapter III discusses how image smoothing and bias reduction are used jointly to improve the performance of image registration algorithms [3, 22, 50]. In Chapter III it was shown that the smoothing portion of the filtering could be accomplished using a low-pass filter to eliminate noise. The chapter also proposed that bias reduction in the algorithm could be performed optically and simulated this optical filtering in experiments. The chapter did not attempt to quantify the exact parameters for the defocus that would be required to perform this optical filtering. This presents another opportunity for future research.

Based on the research in this chapter, it is possible to design a two-lens system (and possibly systems using more than two lenses) for motion estimation that could effectively create two images that, when differenced, would produce a bias-free image that could be reliably registered with a fast correlation-based algorithm. If sensor noise is a concern, the differencing of the image and its optically low-pass filtered content would effectively double the amount of noise that would need to be mitigated. This optical differencing, as was noted in Chapter III also produces images that have

most of their power concentrated in the edges of the objects in the scene. This is also an area that may be fruitful for future exploration.

Another area that could be studied in the future is the analysis and mitigation of bias in different image registration algorithms. Robinson and Milanfar discuss estimator bias for gradient-based image registration in [46]. There are, however, a number of different ways to register images including methods based on cross-correlations as described in Chapter III, methods based on mutual information [43], and methods based on landmarks within images [27]. A comparison of these registration methods and their inherent estimator biases would be an excellent starting point for another dissertation on image registration.

6.2.2 Bounds on Registration Performance. A final area for future research is the possibility of employing the Barankin bound in the area of automatic target recognition (ATR). One of the interesting facets of the Barankin bound that is used in Chapter IV is that it may be used to measure the bound on estimating a shift given other shift scenarios that are slightly different and represent the most probable sources of shift error. In this dissertation, this was used to explore registration estimates of projected images; however, it may also be possible to extend this work to automatic target recognition. For example, Driggers *et al.* [20] test the ability of a group of human test subjects to differentiate between several similar armored vehicles under different noise conditions and sampling rates. Using the Barankin bound, it may be possible to calculate information-theoretic bounds on the ability to differentiate between several similar-looking targets that represent the most probable errors to a target recognition problem. This should be a relatively uncomplicated extension to this work but one that may provide a new way of looking at the ATR problem.

This work on the theoretical bound on image registration performance can also be expanded to include more dynamic cases such as the registration of images under

conditions of rotation and dilation. For a given scene, using the linear operator notation that used in this dissertation, a dilation or a rotation can be modelled as yet another filtering and sampling operation. These bounds have been explored to some extent in [55], however, it may be useful to examine the Barankin bound of images under these same conditions. In fact, because most imaging sensors are square arrays of detectors, any rotation of an imaging sensor necessarily changes the amount of mutual information in two images of the same scene. For some scenarios, this effect may be negligible; however, for remote sensing applications, this change in information may make a substantial difference. Bounds on this type of estimation are another area that provide an opportunity for additional research with military applications.

6.2.3 Block-Based Denoising. As mentioned in Chapter V, the smoothing algorithms from the literature and those introduced in this dissertation all generally produce results that exceed the CRLB. Block-based or other smoothing algorithms provide estimates that are, for most images, much better than a straight forward CRLB calculation would predict by introducing additional assumptions about the frequency content of natural images. In particular, the underlying assumption is that most natural images contain predominantly low-frequency spatial content and that by averaging using linear or non-linear filters, a reasonable estimate of the images under study can be provided.

One interesting potential extension to this research is a calculation of the theoretical bounds on the performance of image smoothing algorithms. One way to approach this problem would be to introduce additional assumptions about image content by modeling the diffraction-limited image as something like a Gibbs distribution [48]. Alternatively, it may be possible to model the image by modeling its local variation using techniques such as those described in [1, 5, 6, 8, 13] and to calculate a bound from this localized structure. Since all of these methods assume that there are local image variations and attempt to account for them, it may be possible to

create either an algorithm-specific or generalized bound for denoising a particular image.

Appendix A. Important Derivations

This appendix provides the derivations of important results used in this dissertation.

A.1 Calculation of the FOM Used in Chapter III

Cain *et al.* in [11] introduced a figure of merit (FOM) which was modified in (3.10) to account for the effects of filtering in the projections. This appendix shows how (3.10) can be used to derive (3.12). For a given image, the numerator and the denominator of (3.10) are examined separately.

The derivation below generalizes some of the random variables using the notation $\mathcal{N}_n(0, K)$ to indicate a Gaussian random variable with zero mean and variance K . Since some of these random variables will be combined in the course of the derivation, the numeric subscript $n \in \{1, 2\}$ is used to indicate the frame of data associated with it. This allows tracking independence of random variables as they are combined to achieve the desired results.

Turning first to the numerator, if the projections of two images are examined over a number of trials, the ensemble average of the cross correlation of these two filtered projections can be written as

$$\mathbf{p}_z = (\mathbf{h}_w * \mathbf{i}_y - w\bar{\mathbf{i}}_y) * \mathbf{W}_f(\mathbf{h}_w * \mathbf{i}_y - w\bar{\mathbf{i}}_y). \quad (\text{A.1})$$

Points on the projection corresponding to the precise alignment of the two filtered projections and a shift of 1 can be expressed as

$$\begin{aligned} \langle \mathbf{p}_z(0) \rangle &= (\mathbf{h}_w * \mathbf{i}_y - w\bar{\mathbf{i}}_y)^T \mathbf{W}_f(\mathbf{h}_w * \mathbf{i}_y - w\bar{\mathbf{i}}_y), \\ \langle \mathbf{p}_z(1) \rangle &= (\mathbf{h}_w * \mathbf{i}_y - w\bar{\mathbf{i}}_y)^T \mathbf{W}_f(\mathbf{h}_w * \mathbf{T}_\alpha \mathbf{i}_y - w\overline{\mathbf{T}_\alpha \mathbf{i}_y})|_{\alpha=1}, \end{aligned} \quad (\text{A.2})$$

where circular shifting is assumed. If $\{L = \sum_{n=1}^N \mathbf{W}_f(n, n)\}$, the difference between these two points can be written as

$$\begin{aligned} \langle \mathbf{p}_z(0) \rangle - \langle \mathbf{p}_z(1) \rangle &= (\mathbf{h}_w * \mathbf{i}_y - w\bar{\mathbf{i}}_y)^T \mathbf{W}_f (\mathbf{h}_w * (\mathbf{i}_y - \mathbf{T}_\alpha \mathbf{i}_y) - w\bar{\mathbf{i}}_y - w\overline{\mathbf{T}_\alpha \mathbf{i}_y}) \Big|_{\alpha=1}, \\ &\approx \frac{L}{N} (\mathbf{h}_w * \mathbf{i}_y - w\bar{\mathbf{i}}_y)^T (\mathbf{h}_w * (\mathbf{i}_y - \mathbf{T}_\alpha \mathbf{i}_y) - w\bar{\mathbf{i}}_y - w\overline{\mathbf{T}_\alpha \mathbf{i}_y}) \Big|_{\alpha=1}. \end{aligned} \quad (\text{A.3})$$

Noting that filtering correlates adjacent terms in the filtered projections and again assuming circular shifting, this can be further reduced to

$$\begin{aligned} \langle \mathbf{p}_z(0) \rangle - \langle \mathbf{p}_z(1) \rangle &\approx \frac{L}{N} (\|\mathbf{i}_y\|^2 - \langle \mathbf{i}_y, \mathbf{T}_\alpha \mathbf{i}_y \rangle \\ &\quad - \langle \mathbf{h}_w * \mathbf{i}_y, w\bar{\mathbf{i}}_y \rangle + \langle \mathbf{h}_w * \mathbf{i}_y, w\overline{\mathbf{T}_\alpha \mathbf{i}_y} \rangle \\ &\quad + \langle w\bar{\mathbf{i}}_y, w\bar{\mathbf{i}}_y \rangle - \langle w\bar{\mathbf{i}}_y, w\overline{\mathbf{T}_\alpha \mathbf{i}_y} \rangle) \Big|_{\alpha=w}, \\ &\approx \frac{L}{N} (\|\mathbf{i}_y\|^2 - \langle \mathbf{i}_y, \mathbf{T}_\alpha \mathbf{i}_y \rangle - \langle w\bar{\mathbf{i}}_y, w\bar{\mathbf{i}}_y \rangle \\ &\quad + \langle w\bar{\mathbf{i}}_y, w\overline{\mathbf{T}_\alpha \mathbf{i}_y} \rangle + \langle w\bar{\mathbf{i}}_y, w\bar{\mathbf{i}}_y \rangle \\ &\quad - \langle w\bar{\mathbf{i}}_y, w\overline{\mathbf{T}_\alpha \mathbf{i}_y} \rangle) \Big|_{\alpha=w}, \\ &\approx L (\text{VAR}[\mathbf{i}] - \overline{\text{COV}}(\mathbf{i}^T \mathbf{T}_\alpha \mathbf{i})) \Big|_{\alpha=w}, \\ &\approx LN^2 (\text{VAR}[\mathbf{I}] - \overline{\text{COV}}(\mathbf{I}|w)). \end{aligned} \quad (\text{A.4})$$

In the denominator of (3.10), the effect of noise on the FOM is accounted for. To do this, the effect of the addition of AWGN of variance σ^2 on the expected value of the variance of the difference between two points on the cross correlation of the two projections is examined. The data in the projection is represented as $\mathbf{d}_{n,z} = \mathbf{i}_y + \mathbf{q}_n$ where \mathbf{q}_n is a vector of Gaussian random variables $\mathcal{N}_n(0, \sigma^2 N)$. Using this

formulation, and again assuming circular shifting,

$$\begin{aligned}
\mathbb{E}[\text{VAR}[\mathbf{p}_z(0) - \mathbf{p}_z(1)]] &= \mathbb{E}[\text{VAR}[(\mathbf{h}_w * (\mathbf{i}_y + \mathbf{q}_1) - w\overline{\mathbf{d}_{1,z}})^T \\
&\quad \mathbf{W}_f((\mathbf{i}_y - \mathbf{T}_\alpha \mathbf{i}_y) + (\mathbf{q}_2 + \mathbf{T}_\alpha \mathbf{q}_2))]]|_{\alpha=w}, \\
&= \text{VAR} \left[\sum_{n=0}^{N-1} \mathbf{W}_f(n, n) (\mathcal{N}_1(0, w\sigma^2 N) \langle \mathbf{i}_y - (\mathbf{T}_\alpha \mathbf{i}_y) \rangle \right. \\
&\quad \left. + \mathcal{N}_1(0, w\sigma^2 N) \mathcal{N}_2(0, 2\sigma^2 N) \right. \\
&\quad \left. + \langle (\mathbf{h}_w * \mathbf{i}_y) - w\overline{\mathbf{d}_{1,z}} \rangle \mathcal{N}_2(0, 2\sigma^2 N) \right] |_{\alpha=w}, \\
&\approx \text{VAR} [\mathcal{N}_1(0, w\sigma^2 LN \langle (\mathbf{i} - (\mathbf{T}_\alpha \mathbf{i}))^2 \rangle) + \mathcal{N}(0, 2w\sigma^4 LN^2)] , \\
&\approx w\sigma^2 LN^2 (2(\text{VAR}(\mathbf{I}) - \overline{\text{COV}}(\mathbf{I}|w)) + \sigma^2) \tag{A.5}
\end{aligned}$$

Combining (A.4) and (A.5) and normalizing by the size of the filter, w , yields the desired result

$$F_{P_y}(0, -1) = \frac{(LN^2 (\text{VAR}[\mathbf{I}] - \overline{\text{COV}}(\mathbf{I}|w)))^2}{w^2\sigma^2 LN^2 (2(\text{VAR}(\mathbf{I}) - \overline{\text{COV}}(\mathbf{I}|w)) + \sigma^2)} \tag{A.6}$$

A.2 Derivation of Theoretical Performance Bounds

A.2.1 Derivation of the CRLB for a Projected & Filtered Image. For a given image, the terms (4.9), (4.10) and (4.11) are derived as follows:

1. Taking the natural logarithm of (4.7) yields

$$\begin{aligned}
\ln p(\mathbf{f}_{1,y}, \mathbf{f}_{2,y} | \mathbf{i}_y, \alpha) &= -N \ln[2\pi\sigma^2 N] - \frac{(\mathbf{f}_{1,y} - \mathbf{i}_y)^T \mathbf{W}^{-1} (\mathbf{f}_{1,y} - \mathbf{i}_y)}{2\sigma^2 N} \\
&\quad - \frac{(\mathbf{f}_{2,y} - \mathbf{H}_p \mathbf{T}_\alpha \mathbf{i}_y)^T \mathbf{W}^{-1} (\mathbf{f}_{2,y} - \mathbf{H}_p \mathbf{T}_\alpha \mathbf{i}_y)}{2\sigma^2 N}. \tag{A.7}
\end{aligned}$$

Taking the partial derivative with respect to α produces

$$\frac{\partial \ln p(\mathbf{f}_{1,y}, \mathbf{f}_{2,y} | \mathbf{i}_y, \alpha)}{\partial \alpha} = -\frac{1}{\sigma^2 N} (\mathbf{f}_{2,y} - \mathbf{H}_p \mathbf{T}_\alpha \mathbf{i}_y)^T \mathbf{W}^{-1} \frac{\partial \mathbf{H}_p \mathbf{T}_\alpha \mathbf{i}_y}{\partial \alpha}. \tag{A.8}$$

Then,

$$\begin{aligned} \frac{\partial^2 \ln p(\mathbf{f}_{1,y}, \mathbf{f}_{2,y} | \mathbf{i}_y, \alpha)}{\partial \alpha^2} &= -\frac{1}{\sigma^2 N} \left((\mathbf{f}_{2,y} - \mathbf{H} \mathbf{T}_\alpha \mathbf{i}_y)^T \mathbf{W}^{-1} \frac{\partial^2 \mathbf{H}_\mathbf{p} \mathbf{T}_\alpha \mathbf{i}_y}{\partial \alpha^2} \right. \\ &\quad \left. - \left(\frac{\partial \mathbf{H}_\mathbf{p} \mathbf{T}_\alpha \mathbf{i}_y}{\partial \alpha} \right)^T \mathbf{W}^{-1} \left(\frac{\partial \mathbf{H}_\mathbf{p} \mathbf{T}_\alpha \mathbf{i}_y}{\partial \alpha} \right) \right). \quad (\text{A.9}) \end{aligned}$$

Using the differentiation property of the Fourier Transform, it can be shown that $\frac{\partial}{\partial \alpha} \mathbf{i}(x - \alpha) = -\frac{\partial}{\partial x} \mathbf{i}(x - \alpha)$ and $\frac{\partial^2}{\partial \alpha^2} \mathbf{i}(x - \alpha) = \frac{\partial^2}{\partial x^2} \mathbf{i}(x - \alpha)$. Using this relationship, the differentiation can be changed to

$$\begin{aligned} \frac{\partial^2 \ln p(\mathbf{f}_{1,y}, \mathbf{f}_{2,y} | \mathbf{i}_y, \alpha)}{\partial \alpha^2} &= -\frac{1}{\sigma^2 N} \left((\mathbf{f}_{2,y} - \mathbf{H}_\mathbf{p} \mathbf{T}_\alpha \mathbf{i}_y)^T \mathbf{W}^{-1} \frac{\partial^2 \mathbf{H}_\mathbf{p} \mathbf{T}_\alpha \mathbf{i}_y}{\partial x^2} \right. \\ &\quad \left. - \left(\frac{\partial \mathbf{H}_\mathbf{p} \mathbf{T}_\alpha \mathbf{i}_y}{\partial x} \right)^T \mathbf{W}^{-1} \left(\frac{\partial \mathbf{H}_\mathbf{p} \mathbf{T}_\alpha \mathbf{i}_y}{\partial x} \right) \right). \quad (\text{A.10}) \end{aligned}$$

Taking the negative of the expectation, it is found that

$$-\mathbb{E} \left[\frac{\partial^2 \ln p(\mathbf{f}_{1,y}, \mathbf{f}_{2,y} | \mathbf{i}_y, \alpha)}{\partial \alpha^2} \right] = \frac{1}{\sigma^2 N} \left(\frac{\partial \mathbf{H}_\mathbf{p} \mathbf{T}_\alpha \mathbf{i}_y}{\partial x} \right)^T \mathbf{W}^{-1} \left(\frac{\partial \mathbf{H}_\mathbf{p} \mathbf{T}_\alpha \mathbf{i}_y}{\partial x} \right).$$

Recalling that $\mathbf{W}^{-1} = (\mathbf{H}_\mathbf{p}^T \mathbf{H}_\mathbf{p})^{-1} = (\mathbf{H}_\mathbf{p})^{-1} (\mathbf{H}_\mathbf{p}^T)^{-1}$ and employing the commutative operator \mathcal{D}_x leads to

$$\begin{aligned} -\mathbb{E} \left[\frac{\partial^2 \ln p(\mathbf{f}_{1,y}, \mathbf{f}_{2,y} | \mathbf{i}_y, \alpha)}{\partial \alpha^2} \right] &= \frac{1}{\sigma^2 N} \|\mathbf{H}_\mathbf{p}^{-1} \mathbf{H}_\mathbf{p} \mathbf{T}_\alpha \mathcal{D}_x \mathbf{i}_y\|^2, \\ &= \frac{1}{\sigma^2 N} \|\mathcal{D}_x \mathbf{i}_y\|^2. \quad (\text{A.11}) \end{aligned}$$

2. Equation (A.8) is then employed to find the elements of the vector $\mathbf{J}_{\alpha\mathbf{x}}$ as:

$$\begin{aligned}
\mathbf{J}_{\alpha\mathbf{x}} &= -\mathbb{E} \left[\frac{\partial}{\partial \mathbf{i}_y} \left(\frac{-1}{\sigma^2 N} (\mathbf{f}_{2,y} - \mathbf{H}_p \mathbf{T}_\alpha \mathbf{i}_y)^T \mathbf{W}^{-1} \frac{\partial \mathbf{H}_p \mathbf{T}_\alpha \mathbf{i}_y}{\partial \alpha} \right) \right] \\
&= \mathbb{E} \left[\frac{1}{\sigma^2 N} \left((\mathbf{f}_{2,y} - \mathbf{H}_p \mathbf{T}_\alpha \mathbf{i}_y)^T \mathbf{W}^{-1} \frac{\partial^2 \mathbf{H}_p \mathbf{T}_\alpha \mathbf{i}_y}{\partial \alpha \partial \mathbf{i}_y} \right. \right. \\
&\quad \left. \left. - \left(\frac{\partial \mathbf{H}_p \mathbf{T}_\alpha \mathbf{i}_y}{\partial \mathbf{i}_y} \right)^T \mathbf{W}^{-1} \frac{\partial \mathbf{H}_p \mathbf{T}_\alpha \mathbf{i}_y}{\partial \alpha} \right) \right], \\
&= \frac{1}{\sigma^2 N} (\mathbf{H}_p \mathbf{T}_\alpha)^T \mathbf{W}^{-1} \left(-\frac{\partial \mathbf{H}_p \mathbf{T}_\alpha \mathbf{i}_y}{\partial \alpha} \right), \\
&= -\frac{1}{\sigma^2 N} \mathbf{T}_\alpha^T \mathbf{H}_p^T (\mathbf{H}_p^T)^{-1} \mathbf{H}_p^{-1} \mathbf{H}_p \mathbf{T}_\alpha \mathcal{D}_x \mathbf{i}_y, \\
&= -\frac{1}{\sigma^2 N} \mathcal{D}_x \mathbf{i}_y. \tag{A.12}
\end{aligned}$$

3. Finally, differentiation with respect to the nuisance parameters themselves produces $\mathbf{J}_{\mathbf{x}\mathbf{x}}$

$$\begin{aligned}
\mathbf{J}_{\mathbf{x}\mathbf{x}} &= -\mathbb{E} \left[\frac{\partial^2}{\partial \mathbf{i}_y^2} \left(-N \ln[2\pi\sigma^2 N] - \frac{(\mathbf{f}_{1,y} - \mathbf{H}_p \mathbf{i}_y)^T \mathbf{W}^{-1} (\mathbf{f}_{1,y} - \mathbf{H}_p \mathbf{i}_y)}{2\sigma^2 N} \right. \right. \\
&\quad \left. \left. - \frac{(\mathbf{f}_{2,y} - \mathbf{H}_p \mathbf{T}_\alpha \mathbf{i}_y)^T \mathbf{W}^{-1} (\mathbf{f}_{2,y} - \mathbf{H}_p \mathbf{T}_\alpha \mathbf{i}_y)}{2\sigma^2 N} \right) \right], \\
&= -\mathbb{E} \left[\frac{\partial}{\partial \mathbf{i}_y} \left(\frac{2\mathbf{H}_p^T \mathbf{W}^{-1} (\mathbf{f}_{1,y} - \mathbf{H}_p \mathbf{i}_y)}{2\sigma^2 N} + \frac{2(\mathbf{H}_p \mathbf{T}_\alpha)^T \mathbf{W}^{-1} (\mathbf{f}_{2,y} - \mathbf{H}_p \mathbf{T}_\alpha \mathbf{i}_y)}{2\sigma^2 N} \right) \right], \\
&= \frac{1}{\sigma^2 N} (\mathbf{H}_p^T \mathbf{W}^{-1} \mathbf{H}_p + \mathbf{T}_\alpha^T \mathbf{H}_p^T \mathbf{W}^{-1} \mathbf{H}_p \mathbf{T}_\alpha), \\
&= \left(\frac{2\mathcal{I}}{\sigma^2 N} \right). \tag{A.13}
\end{aligned}$$

where \mathcal{I} is the identity matrix.

Using (A.11), (A.12), and (A.13) it is now a simple matter to block invert (4.8) using the Schur information complements $\mathbf{S}_\mathbf{x}$ and \mathbf{S}_α as

$$\mathbf{J}^{-1} = \begin{pmatrix} \mathbf{S}_\alpha^{-1} & \mathbf{J}_{\alpha\alpha}^{-1} \mathbf{J}_{\alpha\mathbf{x}} \mathbf{S}_\mathbf{x}^{-1} \\ \mathbf{S}_\mathbf{x}^{-1} \mathbf{J}_{\alpha\mathbf{x}}^T \mathbf{J}_{\alpha\alpha}^{-1} & \mathbf{S}_\mathbf{x}^{-1} \end{pmatrix} \tag{A.14}$$

where

$$\mathbf{S}_\alpha = \mathbf{J}_{\alpha\alpha} - \mathbf{J}_{\alpha\mathbf{x}}\mathbf{J}_{\mathbf{x}\mathbf{x}}^{-1}\mathbf{J}_{\alpha\mathbf{x}}^T, \quad (\text{A.15})$$

$$\mathbf{S}_\mathbf{x} = \mathbf{J}_{\mathbf{x}\mathbf{x}} - \mathbf{J}_{\alpha\mathbf{x}}^T\mathbf{J}_{\alpha\alpha}^{-1}\mathbf{J}_{\alpha\mathbf{x}}. \quad (\text{A.16})$$

Since the purpose of the derivation is to find the CRLB of the shift estimate, the other elements of the FIM can be ignored to solve

$$\mathbf{S}_\alpha^{-1} = \frac{\sigma^2 N}{\|\mathcal{D}_x \mathbf{i}_y\|^2 - \frac{1}{2} \langle \mathcal{D}_x \mathbf{i}_y, \mathcal{D}_x \mathbf{i}_y \rangle}, \quad (\text{A.17})$$

$$= 2\sigma^2 N [\|\mathcal{D}_x \mathbf{i}_y\|^2]^{-1}. \quad (\text{A.18})$$

A.2.2 Derivation of the Two-Dimensional CRLBs. For the 2-D case, the natural logarithm of the joint probability of two image frames represented as vectors (1.8) and (1.9) can be written as

$$\begin{aligned} \ln p(\mathbf{D}_1, \mathbf{D}_2 | \mathbf{I}) &= -\frac{1}{2\sigma^2} (\mathbf{D}_1 - \mathbf{H}_\mathbf{p} \mathbf{H}_\mathbf{o} \mathbf{I})^T \mathbf{W}^{-1} (\mathbf{D}_1 - \mathbf{H}_\mathbf{p} \mathbf{H}_\mathbf{o} \mathbf{I}) \\ &\quad -\frac{1}{2\sigma^2} (\mathbf{D}_2 - \mathbf{H}_\mathbf{p} \mathbf{H}_\mathbf{o} \mathbf{T}_{\alpha,\beta} \mathbf{I})^T \mathbf{W}^{-1} (\mathbf{D}_2 - \mathbf{H}_\mathbf{p} \mathbf{H}_\mathbf{o} \mathbf{T}_{\alpha,\beta} \mathbf{I}) \\ &\quad + \text{constant}. \end{aligned} \quad (\text{A.19})$$

Taking the partial derivative of the log likelihood equation (A.19) with respect to α produces

$$\frac{\partial \ln p(\mathbf{D}_1, \mathbf{D}_2 | \mathbf{I}, \alpha, \beta)}{\partial \alpha} = \frac{1}{\sigma^2} (\mathbf{D}_2 - \mathbf{H}_\mathbf{p} \mathbf{H}_\mathbf{o} \mathbf{T}_{\alpha,\beta} \mathbf{I})^T \mathbf{W}^{-1} \frac{\partial \mathbf{H}_\mathbf{p} \mathbf{H}_\mathbf{o} \mathbf{T}_{\alpha,\beta} \mathbf{I}}{\partial \alpha}. \quad (\text{A.20})$$

Then, the differentiation property of the Fourier Transform is again employed, (i.e. $\frac{\partial}{\partial \alpha} \mathbf{i}(x - \alpha) = -\frac{\partial}{\partial x} \mathbf{i}(x - \alpha)$ and $\frac{\partial^2}{\partial \alpha^2} \mathbf{i}(x - \alpha) = \frac{\partial^2}{\partial x^2} \mathbf{i}(x - \alpha)$) to change the differentiation

to

$$\begin{aligned} \frac{\partial^2 \ln p(\mathbf{D}_1, \mathbf{D}_2 | \mathbf{I}, \alpha, \beta)}{\partial \alpha^2} &= \frac{1}{\sigma^2} \left(\mathbf{D}_2^T \mathbf{W}^{-1} \frac{\partial^2 \mathbf{H}_p \mathbf{H}_o \mathbf{T}_{\alpha, \beta} \mathbf{I}}{\partial \alpha^2} \right. \\ &\quad - \left(\frac{\partial \mathbf{H}_p \mathbf{H}_o \mathbf{T}_{\alpha, \beta} \mathbf{I}}{\partial x} \right)^T \mathbf{W}^{-1} \left(\frac{\partial \mathbf{H}_p \mathbf{H}_o \mathbf{T}_{\alpha, \beta} \mathbf{I}}{\partial x} \right) \\ &\quad \left. - (\mathbf{H}_p \mathbf{H}_o \mathbf{T}_{\alpha, \beta} \mathbf{I})^T \mathbf{W}^{-1} \frac{\partial^2 \mathbf{H}_p \mathbf{H}_o \mathbf{T}_{\alpha, \beta} \mathbf{I}}{\partial \alpha^2} \right). \quad (\text{A.21}) \end{aligned}$$

Taking the negative expectation leads to

$$-\mathbb{E} \left[\frac{\partial^2 \ln p(\mathbf{D}_1, \mathbf{D}_2 | \mathbf{I}, \alpha, \beta)}{\partial \alpha^2} \right] = \frac{1}{\sigma^2} \|\mathcal{D}_x \mathbf{H}_o \mathbf{I}\|^2. \quad (\text{A.22})$$

Similarly,

$$-\mathbb{E} \left[\frac{\partial^2 \ln p(\mathbf{D}_1, \mathbf{D}_2 | \mathbf{I}, \alpha, \beta)}{\partial \beta^2} \right] = \frac{1}{\sigma^2} \|\mathcal{D}_y \mathbf{H}_o \mathbf{I}\|^2. \quad (\text{A.23})$$

Then, beginning with (A.20)

$$\begin{aligned} \frac{\partial^2 \ln p(\mathbf{D}_1, \mathbf{D}_2 | \mathbf{I}, \alpha, \beta)}{\partial \alpha \partial \beta} &= \frac{1}{\sigma^2} \left[\mathbf{D}_2^T \mathbf{W}^{-1} \frac{\partial^2 \mathbf{H}_p \mathbf{H}_o \mathbf{T}_{\alpha, \beta} \mathbf{I}}{\partial \alpha \partial \beta} \right. \\ &\quad - \left(\frac{\partial \mathbf{H}_p \mathbf{H}_o \mathbf{T}_{\alpha, \beta} \mathbf{I}}{\partial \beta} \right)^T \mathbf{W}^{-1} \frac{\partial \mathbf{H}_p \mathbf{H}_o \mathbf{T}_{\alpha, \beta} \mathbf{I}}{\partial \alpha} \\ &\quad \left. - (\mathbf{H}_p \mathbf{H}_o \mathbf{T}_{\alpha, \beta} \mathbf{I})^T \mathbf{W}^{-1} \frac{\partial^2 \mathbf{H}_p \mathbf{H}_o \mathbf{T}_{\alpha, \beta} \mathbf{I}}{\partial \alpha \partial \beta} \right]. \quad (\text{A.24}) \end{aligned}$$

Taking the negative expectation and changing the variables of differentiation yields

$$-\mathbb{E} \left[\frac{\partial^2 \ln p(\mathbf{D}_1, \mathbf{D}_2 | \mathbf{I}, x, y)}{\partial \alpha \partial \beta} \right] = \langle \mathcal{D}_x \mathbf{H}_o \mathbf{I}, \mathcal{D}_y \mathbf{H}_o \mathbf{I} \rangle. \quad (\text{A.25})$$

Differentiating with respect to the nuisance parameters, again return to (A.20) and find,

$$\frac{\partial^2 \ln p(\mathbf{D}_1, \mathbf{D}_2 | \mathbf{I}, \alpha, \beta)}{\partial \alpha \partial \mathbf{I}} = \frac{1}{\sigma^2} \left[(\mathbf{H}_p \mathbf{H}_o \mathbf{T}_{\alpha, \beta})^T \mathbf{W}^{-1} \frac{\partial \mathbf{H}_p \mathbf{H}_o \mathbf{T}_{\alpha, \beta} \mathbf{I}}{\partial \alpha} \right]. \quad (\text{A.26})$$

Completing the differentiation, changing the variable of differentiation, and taking the negative expectation produces the results

$$\begin{aligned} \mathbb{E} \left[\frac{\partial^2 \ln p(\mathbf{D}_1, \mathbf{D}_2 | \mathbf{I}, \alpha, \beta)}{\partial \alpha \partial \mathbf{I}} \right] &= -\frac{1}{\sigma^2} (\mathbf{H}_p \mathbf{H}_o \mathbf{T}_{\alpha, \beta})^T \mathbf{W}^{-1} \frac{\partial \mathbf{H}_p \mathbf{H}_o \mathbf{T}_{\alpha, \beta} \mathbf{I}}{\partial x}, \\ &= -\frac{1}{\sigma^2} \mathbf{H}_o^T \mathcal{D}_x \mathbf{H}_o \mathbf{I} \end{aligned} \quad (\text{A.27})$$

and similarly

$$\mathbb{E} \left[\frac{\partial^2 \ln p(\mathbf{D}_1, \mathbf{D}_2 | \mathbf{I}, \alpha, \beta)}{\partial \beta \partial \mathbf{I}} \right] = -\frac{1}{\sigma^2} \mathbf{H}_o^T \mathcal{D}_y \mathbf{H}_o \mathbf{I}. \quad (\text{A.28})$$

Finally, derive

$$\begin{aligned} -\mathbb{E} \left[\frac{\partial^2}{\partial \mathbf{I}^2} \ln p(\mathbf{D}_1, \mathbf{D}_2 | \mathbf{I}) \right] &= \mathbb{E} \left[\frac{1}{\sigma^2} \frac{\partial}{\partial \mathbf{I}} \left[(\mathbf{D}_1 - \mathbf{H}_p \mathbf{H}_o \mathbf{I})^T \mathbf{W}^{-1} (\mathbf{H}_p \mathbf{H}_o) \right. \right. \\ &\quad \left. \left. + (\mathbf{D}_2 - \mathbf{H}_p \mathbf{H}_o \mathbf{T}_{\alpha, \beta} \mathbf{I})^T \mathbf{W}^{-1} (\mathbf{H}_p \mathbf{H}_o \mathbf{T}_{\alpha, \beta}) \right] \right], \\ &= \frac{1}{\sigma^2} \text{diag} \left((\mathbf{H}_p \mathbf{H}_o)^T \mathbf{W}^{-1} \mathbf{H}_p \mathbf{H}_o \right. \\ &\quad \left. + (\mathbf{H}_p \mathbf{H}_o \mathbf{T}_{\alpha, \beta})^T \mathbf{W}^{-1} (\mathbf{H}_p \mathbf{H}_o \mathbf{T}_{\alpha, \beta}) \right), \\ &= \frac{2}{\sigma^2} \text{diag} (\mathbf{H}_o^T \mathbf{H}_o) \end{aligned} \quad (\text{A.29})$$

If a shift vector is defined $\gamma = [\alpha, \beta]$, a block FIM \mathbf{J} can be created of the form:

$$\mathbf{J} = \begin{pmatrix} \mathbf{J}_{\gamma\gamma} & \mathbf{J}_{\gamma o}^T \\ \mathbf{J}_{\gamma o} & \mathbf{J}_{oo} \end{pmatrix} \quad (\text{A.30})$$

where, for the filtered images, combining results from (A.22), (A.23), and (A.25) gives

$$\mathbf{J}_{\gamma\gamma} = \frac{1}{\sigma^2} \begin{pmatrix} \|\mathcal{D}_x \mathbf{H}_o \mathbf{I}\|^2 & \langle \mathcal{D}_x \mathbf{H}_o \mathbf{I}, \mathcal{D}_y \mathbf{H}_o \mathbf{I} \rangle \\ \langle \mathcal{D}_x \mathbf{H}_o \mathbf{I}, \mathcal{D}_y \mathbf{H}_o \mathbf{I} \rangle & \|\mathcal{D}_y \mathbf{H}_o \mathbf{I}\|^2 \end{pmatrix} \quad (\text{A.31})$$

$$\mathbf{J}_{\gamma o} = -\frac{1}{\sigma^2} \begin{bmatrix} \mathbf{H}_o^T \mathcal{D}_x \mathbf{H}_o \mathbf{I} & \mathbf{H}_o^T \mathcal{D}_y \mathbf{H}_o \mathbf{I} \end{bmatrix} \quad (\text{A.32})$$

$$\mathbf{J}_{oo} = \frac{2}{\sigma^2} \text{diag}(\mathbf{H}_o^T \mathbf{H}_o). \quad (\text{A.33})$$

Inverting the FIM yields

$$\mathbf{J}^{-1} = \begin{pmatrix} \mathbf{S}_\gamma^{-1} & \mathbf{J}_{\gamma\gamma}^{-1} \mathbf{J}_{\gamma o} \mathbf{S}_o^{-1} \\ \mathbf{S}_o^{-1} \mathbf{J}_{\gamma o}^T \mathbf{J}_{\gamma\gamma}^{-1} & \mathbf{S}_o^{-1} \end{pmatrix} \quad (\text{A.34})$$

where

$$\mathbf{S}_\gamma = \mathbf{J}_{\gamma\gamma} - \mathbf{J}_{\gamma o} \mathbf{J}_{oo}^{-1} \mathbf{J}_{\gamma o}^T, \quad (\text{A.35})$$

$$\mathbf{S}_o = \mathbf{J}_{oo} - \mathbf{J}_{\gamma o}^T \mathbf{J}_{\gamma\gamma}^{-1} \mathbf{J}_{\gamma o}. \quad (\text{A.36})$$

Then

$$\mathbf{S}_\gamma^{-1} = 2\sigma^2 \begin{bmatrix} \|\mathcal{D}_x \mathbf{H}_o \mathbf{I}\|^2 & \langle \mathcal{D}_x \mathbf{H}_o \mathbf{I}, \mathcal{D}_y \mathbf{H}_o \mathbf{I} \rangle \\ \langle \mathcal{D}_x \mathbf{H}_o \mathbf{I}, \mathcal{D}_y \mathbf{H}_o \mathbf{I} \rangle & \|\mathcal{D}_y \mathbf{H}_o \mathbf{I}\|^2 \end{bmatrix}^{-1} \quad (\text{A.37})$$

Appendix B. Mathematical Background and Related Theory

This appendix describes some of the basic mathematical concepts and notational conventions used throughout this paper. It describes some of the statistical underpinnings of image denoising problems and describe the notation that is used commonly in the dissertation to address the related problems of block-based image denoising and image registration.

Section B.1 provides a discussion of the chi-square distribution. The section provides a discussion of some of the salient features of the chi-square distribution and a discussion of how it occurs in image denoising problems. This dissertation also uses the correlation structure of an image as an exploitable underlying property of a noisy set of data. Section B.2 describes the notation and methods used for measuring the correlation structure of the projections of an image and of an entire image. The Cramer-Rao lower bound and Barankin bound are two established methods for estimating bounds on the mean-squared error of a parameter such as alignment. Therefore, Section 2.3 reviews the Cramer-Rao lower bound and the Barankin bound. Finally, Section 2.4 describes the method used to model defocus errors in an optical system.

B.1 The Chi-Square Distribution

This section describes the chi-square distribution and provides a working understanding of it that is of fundamental importance in describing and understanding the denoising algorithms in this dissertation. Although a thorough description of the characteristics of the noncentral chi-square distribution is contained in [29] among other places, it is worthwhile to discuss the significance of this distribution within the context of the image denoising problem. As noted in [18], [28], and [29], the noncentral chi-square distribution is the PDF of the sum of normally-distributed, unit-variance, squared random variables x_i with means μ_i . If x is a random variable

with a chi-square distribution, then

$$x = \sum_{i=1}^{\nu} x_i^2. \quad (\text{B.1})$$

In this equation the parameter ν is referred to as the “degrees of freedom” of the variable. The sum of the squares of the means forms the “noncentrality parameter” of the chi-square distribution. This parameter is denoted as λ where

$$\lambda = \sum_{i=1}^{\nu} \mu_i^2. \quad (\text{B.2})$$

The PDF of the distribution may be written

$$p(x) = \frac{x^{\frac{\nu}{2}-1} e^{-\frac{1}{2}(x+\lambda)}}{2^{\nu/2}} \sum_{k=0}^{\infty} \frac{(\frac{\lambda x}{4})^k}{k! \Gamma(-\frac{\nu}{2} + k)}, \quad x \geq 0, \quad (\text{B.3})$$

where the Gamma function, $\Gamma(u)$ is defined as

$$\Gamma(u) = \int_0^{\infty} t^{u-1} e^{-t} dt. \quad (\text{B.4})$$

The noncentral chi-square distribution has the attributes

$$E[x] = \nu + \lambda, \quad (\text{B.5})$$

$$\text{VAR}[x] = 2\nu + 4\lambda. \quad (\text{B.6})$$

This distribution is denoted using the symbol $\chi_{\nu}^2(\lambda)$.

Using (B.3), it will be useful to examine the behavior of the distribution using varying values of ν and λ . These distributions are shown in Figures B.1, B.2, B.3, B.4, and B.5. In the next section, a discussion of the reason for the occurrence of this distribution will be presented. It will become evident that λ is a function of the noise present in the image and will be fixed in the applications presented.

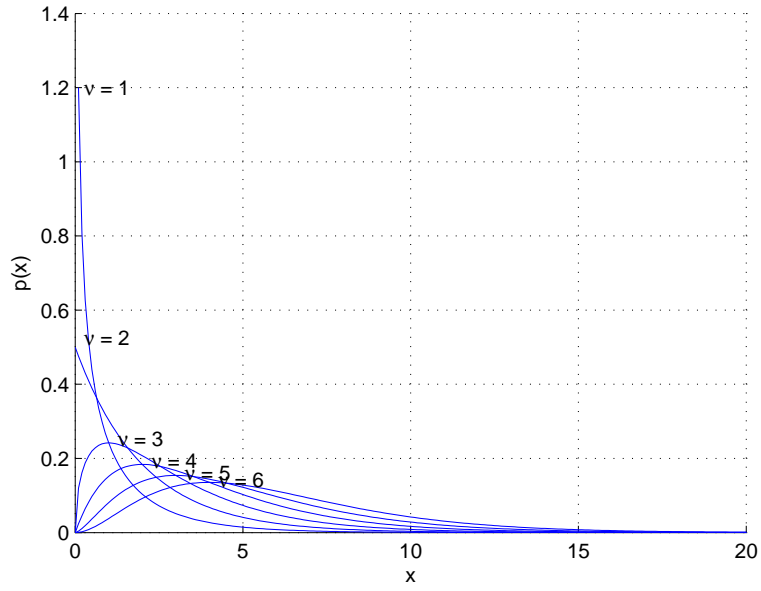


Figure B.1: Graph of the PDF of $\chi^2_\nu(\lambda)$ with $\lambda = 0$. This distribution is identical to that of a central chi-square distribution. Varying values of ν are plotted.

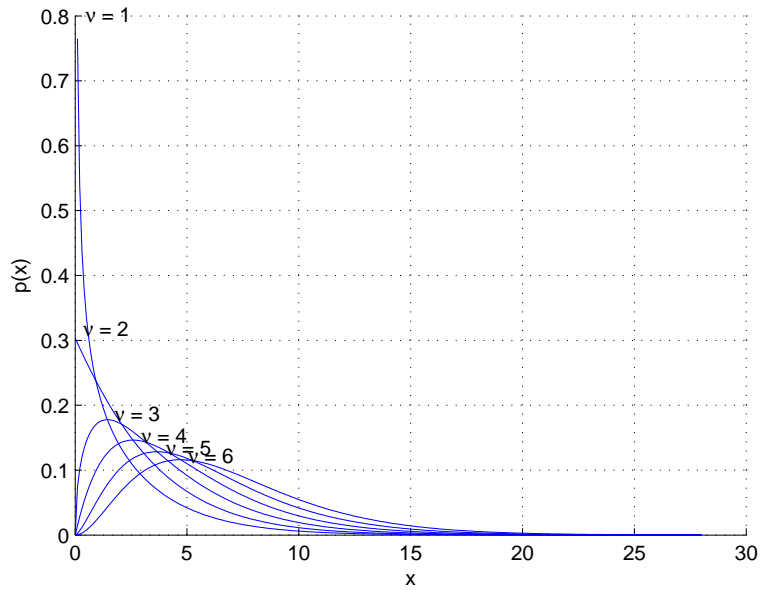


Figure B.2: Graph of the PDF of $\chi^2_\nu(\lambda)$ with $\lambda = 1$. Varying values of ν are plotted.

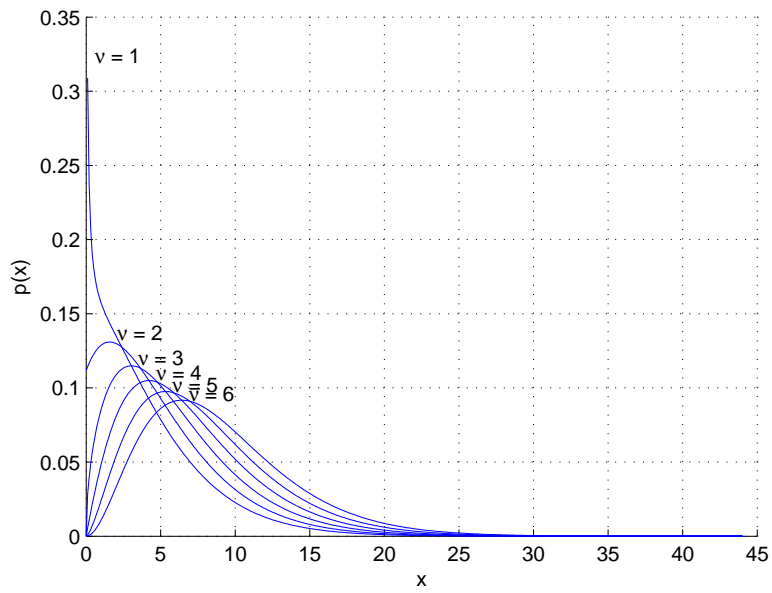


Figure B.3: Graph of the PDF of $\chi^2_\nu(\lambda)$ with $\lambda = 3$. Varying values of ν are plotted.

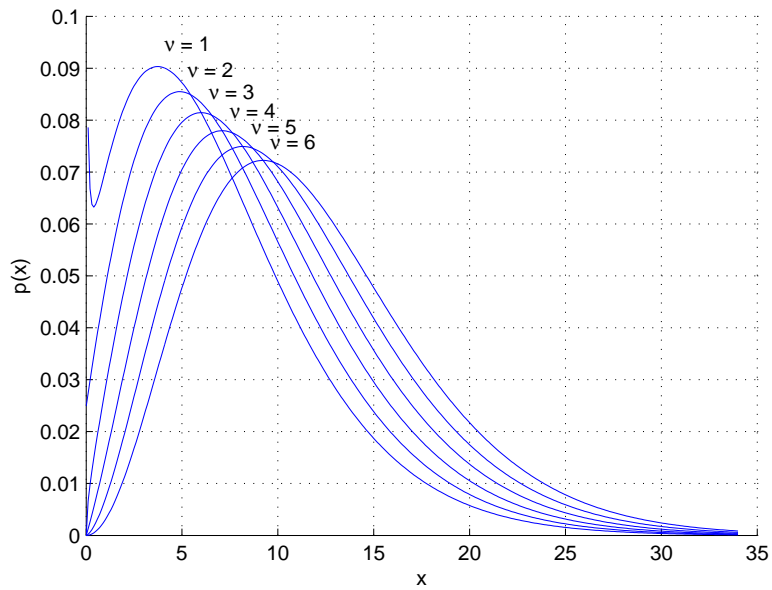


Figure B.4: Graph of the PDF of $\chi^2_\nu(\lambda)$ with $\lambda = 6$. Varying values of ν are plotted.

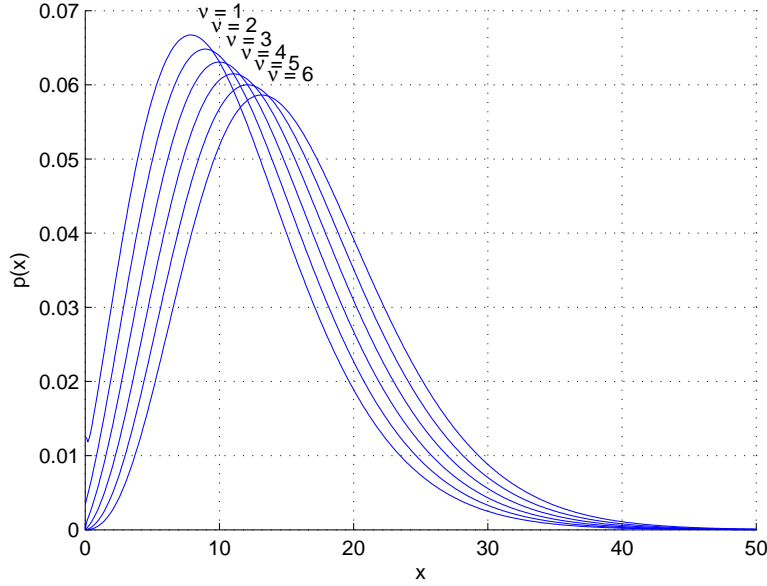


Figure B.5: Graph of the PDF of $\chi_\nu^2(\lambda)$ with $\lambda = 10$. Varying values of ν are plotted.

B.1.1 Occurrence of the Chi-Square Distribution in Image Processing Problems. The research described in this dissertation includes development of methods for denoising images that rely on mean-squared error calculations between image subregions. When one looks at the mean-squared error between identical diffraction-limited subimages that have additive Gaussian noise, the PDF of the resulting mean-squared error distribution will be a noncentral chi-square distribution. The reason that this distribution occurs can be explained using the following logic.

Say \mathbf{I} is defined as a diffraction-limited, $n \times n$ block of an image where n is a positive integer. For notational simplicity, ignore the location of this block within the image and use the variables u and v to index the individual pixels of this subimage. If \mathbf{I} is corrupted by zero-mean Gaussian noise so that for each pixel, $\mathbf{I}(u, v)$ where $u \in \{1, \dots, n\}$ and $v \in \{1, \dots, n\}$, then

$$\mathbf{D}_{i,j}(u, v) = \mathbf{I}(u, v) + \mathbf{Q}_{i,j}(u, v),$$

where $\mathbf{D}_{i,j}(u, v)$ is a subimage centered at (i, j) and $\mathbf{Q}_{i,j}$ is the realization of the Gaussian noise within that subimage. In matrix notation this can be described

$$\mathbf{D}_{i,j} = \mathbf{I} + \mathbf{Q}_{i,j},$$

where $\mathbf{Q}_{i,j}$ is an $n \times n$ matrix of Gaussian random variables and represents the realization of noise in this block. Within an image, there may be other $n \times n$ blocks centered at coordinates (s, t) that satisfy the equation

$$\mathbf{D}_{s,t} = \mathbf{I} + \mathbf{Q}_{s,t}.$$

If these subimages exist, then $\Delta_{s,t}$, the error between $\mathbf{D}_{i,j}$ and $\mathbf{D}_{s,t}$ can be calculated

$$\begin{aligned} \Delta_{s,t} &= \mathbf{D}_{i,j} - \mathbf{D}_{s,t} = (\mathbf{I} + \mathbf{Q}_{i,j}) - (\mathbf{I} + \mathbf{Q}_{s,t}), \\ &= \mathbf{Q}_{i,j} - \mathbf{Q}_{s,t}. \end{aligned} \tag{B.7}$$

For a single pixel in $\Delta_{s,t}$,

$$\Delta_{s,t}(u, v) = \mathbf{Q}_{i,j}(u, v) - \mathbf{Q}_{s,t}(u, v).$$

At this point, note that $\mathbf{Q}_{i,j}(u, v)$ stays constant while $\mathbf{Q}_{s,t}(u, v)$ varies over all the blocks that satisfy the original conditions. Thus it is possible to treat $\mathbf{Q}_{i,j}(u, v)$ as a deterministic value and $\mathbf{Q}_{s,t}(u, v)$ as a Gaussian random variable with standard deviation σ . The PDF of $\Delta_{s,t}(u, v)$ can then be written as

$$p(\Delta_{s,t}(u, v) | \mathbf{Q}_{i,j}(u, v)) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(\Delta_{s,t}(u, v) - \mathbf{Q}_{i,j}(u, v))^2}{\sigma^2}\right). \tag{B.8}$$

For notational simplicity define the Frobenius norm as Watkins does in [54] using the notation $\|\mathbf{A}\|_F$ to be

$$\|\mathbf{A}\|_F = \sqrt{\sum_{i=1}^n \sum_{j=1}^n |\mathbf{A}(i, j)|^2}, \quad (\text{B.9})$$

where $\mathbf{A}(i, j)$ are the elements of the $n \times n$ matrix \mathbf{A} . Using the Frobenius norm, the mean-squared error can then be calculated as the quantity

$$MSE = \frac{\|\Delta_{s,t}\|_F^2}{n^2}, \quad (\text{B.10})$$

where the numerator is a sum over all u and v indexing $\Delta_{s,t}$ and the denominator is a constant that depends on the size of the subimage. Furthermore, the numerator is a sum of the squares of Gaussian random variables of varying means. The sum of these Gaussian random variables is, by definition, a noncentral chi-square distribution [28]. The arrival at this distribution can also be shown experimentally as shown in the following subsection.

B.1.2 Statistical Characteristics of an Experimentally Determined Distribution. Say there is a single 3×3 portion of a diffraction-limited image that can be represented as shown in (B.11). This subimage is denoted as \mathbf{I} .

$$\mathbf{I} = \begin{pmatrix} 100 & 120 & 131 \\ 45 & 190 & 43 \\ 100 & 140 & 100 \end{pmatrix} \quad (\text{B.11})$$

If zero-mean, normally-distributed noise with $\sigma = 25$ is added to \mathbf{I} , a noise realization for the block may be obtained that is similar to $\mathbf{Q}_{i,j}$ shown in (B.12).

$$\mathbf{Q}_{i,j} = \begin{pmatrix} -10.8141 & 7.1919 & 29.7291 \\ -41.6396 & -28.6618 & -0.9408 \\ 3.1333 & 29.7729 & 8.1823 \end{pmatrix} \quad (\text{B.12})$$

Using the notation $\langle \rangle$ to indicate a sample mean, this realization of the noise has a mean of $\langle \mathbf{Q}_{i,j} \rangle = \frac{1}{9} \sum_{u=1}^n \sum_{v=1}^n \mathbf{Q}_{i,j}(u, v) = -0.45$ and a variance of $\sigma^2 = 507.78$.

At this point it is helpful to note that although the distribution of the noise has zero mean and $\sigma^2 = 625$, the mean of the realization of the noise in the 3×3 matrix is not zero and the variance is less than the variance that would be observed over a larger array of numbers. This difference will become significant later on. For now, it is interesting to examine statistics of the mean values of the noise.

B.1.3 Statistics of the Sample Mean of $n \times n$ Noise Samples. For an $n \times n$ sized subimage, the mean of the noise within the subimage is defined

$$\langle \mathbf{Q}_{i,j} \rangle = \frac{\sum_{u=1}^n \sum_{v=1}^n \mathbf{Q}_{i,j}(u, v)}{n^2}. \quad (\text{B.13})$$

In this equation, $\sum_{u=1}^n \sum_{v=1}^n \mathbf{Q}_{i,j}(u, v)$ represents the sum of i.i.d. zero-mean Gaussian random variables. To calculate the PDF of this mean, recall that the PDF of the AWGN can be represented as

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(\frac{-x^2}{2\sigma^2}\right). \quad (\text{B.14})$$

As noted in [35] the sum of random variables with PDFs described by (B.14) can be found as

$$p(u) = \frac{n^2}{\sqrt{2\pi}n\sigma} \left(\frac{-(nu)^2}{2\sigma^2}\right). \quad (\text{B.15})$$

Using (B.15), it can be shown analytically that varying the size of the $n \times n$ subimage will change the variance of the mean of the noise. In Figure B.6 this is demonstrated graphically using analytic data to graph the mean of the noise in blocks of varying size where $\sigma = 2$. It is verified using randomly generated experimental data in Figure B.7.

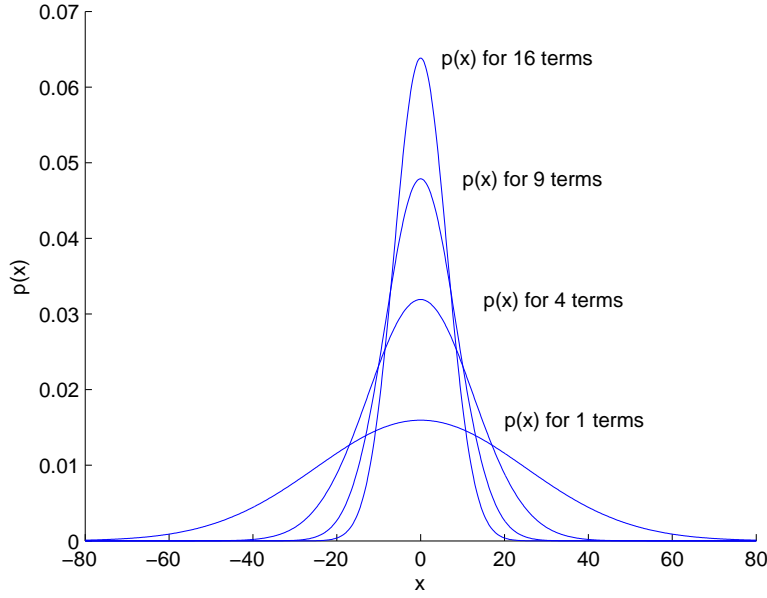


Figure B.6: Analytically constructed plot of the PDF of the mean of a set of noise samples. Note that the sample mean of a set of zero-mean noise realizations is more likely to be zero with a larger block size.

B.1.3.1 Corruption of a Subimage with Noise. Returning again use to the image model $\mathbf{D}_{i,j} = \mathbf{I} + \mathbf{Q}_{i,j}$ a sample noisy block with AWGN can be created as

$$\mathbf{D}_{i,j} = \begin{pmatrix} 89.1859 & 127.1919 & 160.7291 \\ 3.3604 & 161.3382 & 42.0592 \\ 103.1333 & 169.7729 & 108.1823 \end{pmatrix}. \quad (\text{B.16})$$

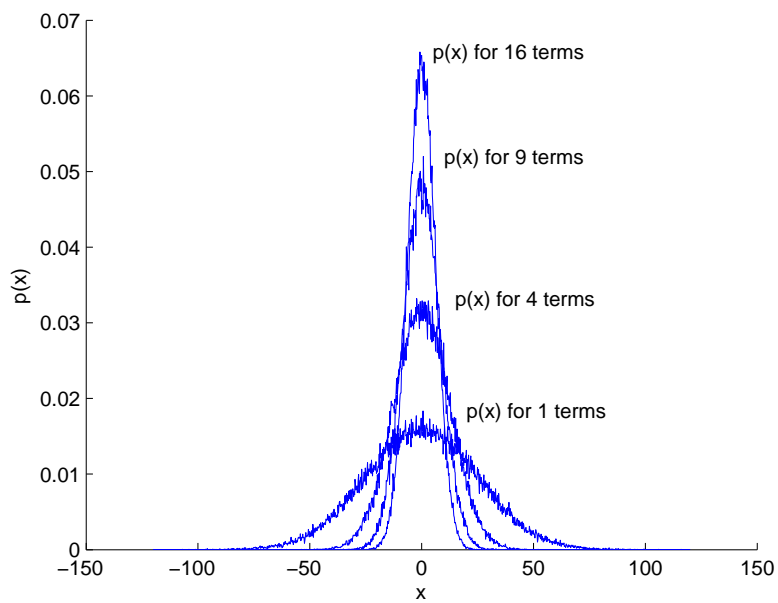


Figure B.7: Experimentally constructed plot of the PDF of the mean of a set of noise samples. Data was constructed using 100,000 ensembles of data with the indicated block size. Results closely approximate those predicted analytically and shown in Figure B.6

The mean of this noisy block is 107.217 and its variance is 2831.0. For reasons that will become evident later in this dissertation, it is desirable like to subtract the mean from this block and then examine its mean-squared error in relation to other blocks that have been created through a similar process. This process will yield a chi-square random variable. If a mean-subtracted block is created $\mathbf{F}_{i,j} = \mathbf{D}_{i,j} - \langle \mathbf{D}_{i,j} \rangle$ the zero-mean result is

$$\mathbf{F}_{i,j} = \begin{pmatrix} -18.0311 & 19.9749 & 53.5121 \\ -103.8566 & 54.1212 & -65.1579 \\ -4.0837 & 62.5559 & 0.9653 \end{pmatrix}. \quad (\text{B.17})$$

This block is now compared to other blocks that are statistically similar.

B.1.3.2 Generating Statistically Similar Subimages. Say that there exist 10,000 blocks that are identical to block \mathbf{I} and say that these blocks are also corrupted by zero-mean Gaussian noise with standard deviation 25. Using the same process that was employed to construct $\mathbf{F}_{i,j}$, subtract the mean from these blocks and call these blocks $\mathbf{G}_{s,t}$ where $s \in \{1, \dots, 100\}$ and $t \in \{1, \dots, 100\}$. Calling $\Delta(u, v)$ the per-pixel error between $\mathbf{F}_{i,j}$ and $\mathbf{G}_{s,t}$ where the pixels are indexed using u, v , the errors observed are shown in Figure B.8.

Assuming there are enough samples available to equate the sample statistics with a true, underlying statistical distribution, define the expected value of the error between $\mathbf{F}_{i,j}$ and $\mathbf{G}_{i,j}$ as $\hat{\Delta}_{s,t}$. Then,

$$\hat{\Delta}_{s,t}(u, v) = \sum_{\Delta_{s,t}(u,v)=-\infty}^{\infty} \Delta_{s,t}(u, v) p(\Delta_{s,t}(u, v)) \quad (\text{B.18})$$

$$\Rightarrow \hat{\Delta}_{s,t} = \begin{pmatrix} -10.3264 & 7.4875 & 30.1458 \\ -40.8786 & -27.8308 & -0.841 \\ 3.48 & 30.5208 & 8.2521 \end{pmatrix}. \quad (\text{B.19})$$

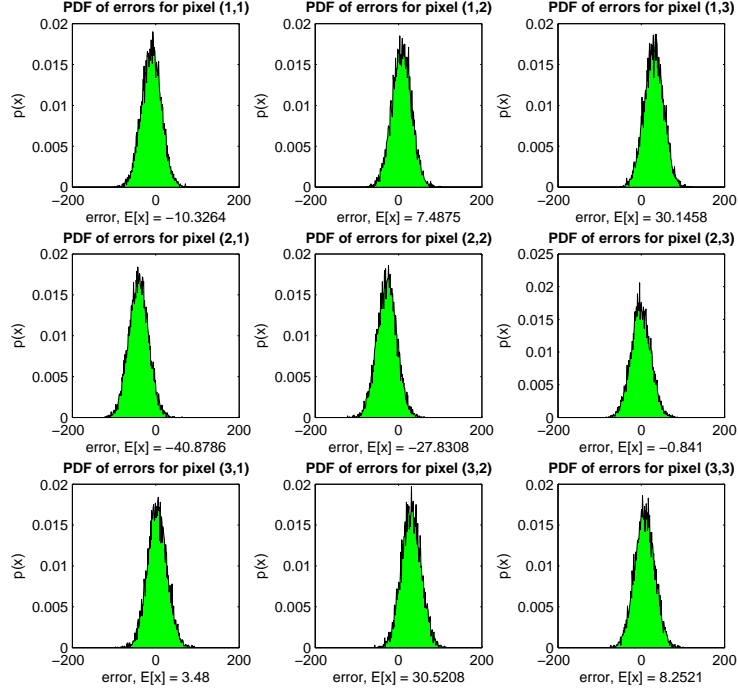


Figure B.8: Per pixel error between similarly constructed images.

These errors are approximately equal to $\mathbf{Q}_{i,j} - \langle \mathbf{Q}_{i,j} \rangle$. The expected value for $\langle \mathbf{Q}_{i,j} \rangle$ is zero, and since *a priori* knowledge of $\mathbf{Q}_{i,j}$ is not available, it is necessary to assume $\langle \mathbf{Q}_{i,j} \rangle = 0$ and $\hat{\Delta}_{s,t}(u, v) = \mathbf{Q}_{i,j}(u, v)$.

B.1.3.3 Estimation of Mean-Squared Error . Under the aforementioned assumptions, the PDF of the measured mean-squared error between \mathbf{I} and $\mathbf{F}_{i,j}$ can be predicted. Since the noncentral chi-square distribution is constructed from normal-variance Gaussian random variables, in order to use the given equations, it is necessary to normalize the distributions. Examining the per-pixel squared error for $\sigma^2 \neq 1$ and $\nu = 9$ the following calculation is made for the chi-square random variable

$$\frac{x}{\sigma^2} = \sum_{i=1}^9 \frac{x_i^2}{\sigma^2}. \quad (\text{B.20})$$

The scaled noncentrality parameter is found

$$\frac{\lambda}{\sigma^2} = \sum_{i=1}^9 \frac{\mu_i^2}{\sigma^2} = \sum_{i=1}^9 \frac{(E[x^2])}{\sigma^2} = \sum_{i=1}^9 1 = 9. \quad (\text{B.21})$$

The expected mean of the scaled data is calculated from the known values for the noncentrality parameter λ and the number of degrees of freedom ν to be

$$\frac{E[x]}{\sigma^2} = \nu + \lambda = 9 + 9 = 18, \quad (\text{B.22})$$

which lead directly to the first moment of the PDF

$$E[x] = 11,250. \quad (\text{B.23})$$

The variance can also be calculated from the noncentrality parameter λ and the number of degrees of freedom ν as

$$\text{VAR}\left[\frac{x}{\sigma^2}\right] = 2\nu + 4\lambda = 2(9) + 4(9) = 54, \quad (\text{B.24})$$

which then leads to the variance

$$\text{VAR}[x] = 54625^2 = 21,093,750.$$

To examine the mean-squared error of variables with nine degrees of freedom (corresponding to the nine pixels in the subimage), divide the mean values by 9 and the variance value by $9^2 = 81$ to give the mean value over each block, thus yielding

$$\begin{aligned} E[x]_{\text{predicted}} &= 1,250, \\ \text{VAR}[x]_{\text{predicted}} &= 260,416. \end{aligned} \quad (\text{B.25})$$

Modifying the calculations to incorporate knowledge of the actual means then,

$$\frac{\lambda}{\sigma^2} = \sum_{u=1}^3 \sum_{v=1}^3 \left(\frac{\hat{E}_{s,t}(u,v)}{\sigma} \right)^2 = 7.247. \quad (\text{B.26})$$

and

$$\begin{aligned} \text{E}[x]_{\text{calculated}} &= 1,127, \\ \text{VAR}[x]_{\text{calculated}} &= 227,120. \end{aligned} \quad (\text{B.27})$$

Using randomly generated data as described above, the measured values are

$$\begin{aligned} \langle x \rangle_{\text{calculated}} &= 1,062, \\ \langle x^2 \rangle_{\text{calculated}} - \langle x \rangle_{\text{calculated}}^2 &= 218,220. \end{aligned} \quad (\text{B.28})$$

Graphically, the measured PDF vs. the predicted and calculated PDFs are shown below in Figure B.9. As shown graphically, the error distribution can be predicted with reasonable accuracy using only the block size and the know mean and standard deviation of the noise.

B.2 Calculating the Covariance Present in Images and Image Projections

Average covariances are used in some calculations in this dissertation. This section describes these covariances are measured and the notation that is used to denote these measurements. In Chapter III, the covariances of image pixels are calculated based on their measured values in projections. Since the calculations rely on information available in the projections, out of necessity, these measures of pixel covariances ignore some relationships that are evident in 2-D that are not evident in 1-D projections. Consequently, some of the sample covariances measured are better described as average covariances. This averaging is acknowledged using overbar notation and expressed as $\overline{\text{COV}}_z$, $z \in \{x, y\}$ as they occur where z indicates

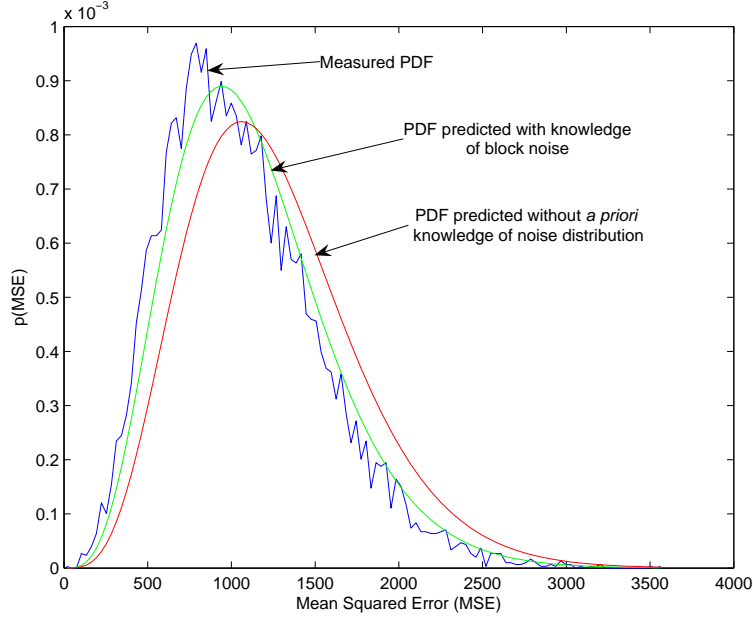


Figure B.9: Comparison of measured vs. predicted data with $\chi^2_{\nu}(\lambda)$ with $\lambda = 15$ distribution. Data labeled “measured PDF” is calculated according to the conditions of (B.29), data labelled “PDF predicted with knowledge of block noise” is calculated according to conditions of (B.28) and data labelled “PDF predicted without *a priori* knowledge” is calculated according to conditions of B.28.

the axis of the projection. For a point indexed with x in a projection define this average covariance as

$$\overline{\text{COV}}_y(\mathbf{I}|\alpha) = \frac{1}{N(N-1)} \left\langle \sum_{y_1=0}^{N-1} \sum_{\substack{y_2=0 \\ y_2 \neq y_1}}^{N-1} \mathbf{I}(x, y_1) \mathbf{I}(x + \alpha, y_2) \right\rangle - \langle \mathbf{I} \rangle^2. \quad (\text{B.29})$$

In (B.29), $\langle \rangle$ indicates an average over all x in the first term and an average over all x and y in the second term. Sample plots of covariances measured for two images are shown in Figures B.10-B.13.

When uncorrelated noise is added to the image, the center of the covariance plot (corresponding to the variance of the noise) increases. However, the effect on



Figure B.10: 1024×1024 grayscale image of the Pentagon image from <http://sipi.usc.edu/database/>.

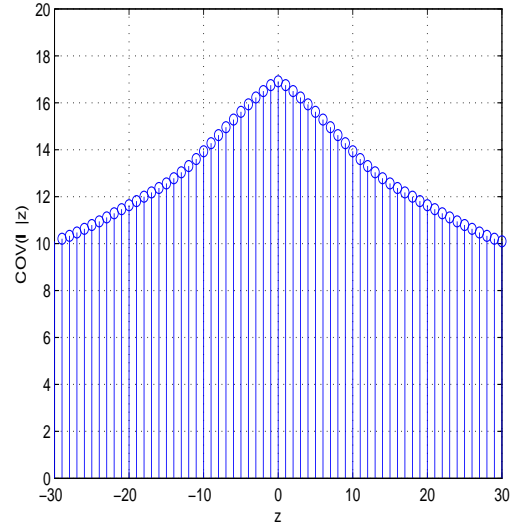


Figure B.11: Measured covariance of the column projections of the image in Figure B.10. The covariance in this graph is calculated using circular shifting.

the off-center values of the covariance function is much less pronounced. This effect is shown pictorially in Figures B.14 and B.15. In fact, the off-peak values are similar enough to suggest that the correlation structure for these images may be estimated fairly accurately by low-pass filtering the measured covariance function of a noisy image and estimating the peak using the slopes of the points $z = -\alpha$ through $z = +\alpha$ where α represents an arbitrary but small value. This will allow the adaptive design of a filter for a given image that will minimize registration errors. It may assist in design a optimal filter to denoise the image.



Figure B.12: 256×256 grayscale aerial image of a chemical plant from <http://sipi.usc.edu/database/>.

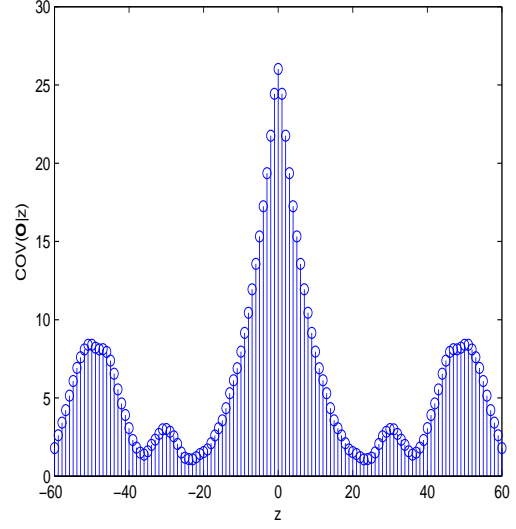


Figure B.13: Measured covariance of the column projections of the image in Figure B.12. The covariance in this graph is calculated using circular shifting.

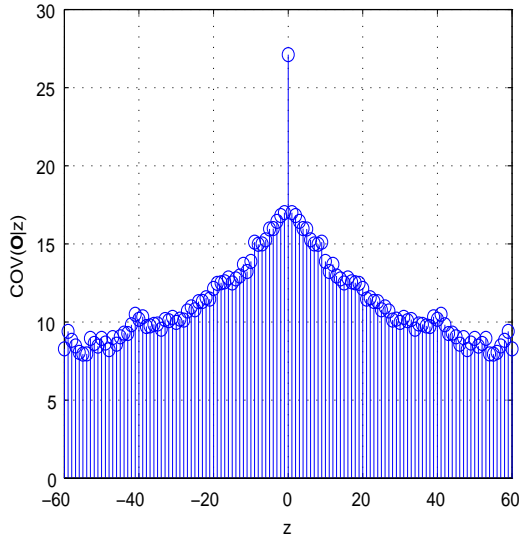


Figure B.14: Measured covariance of column projections of the pentagon image with AWGN of $\sigma = 100$.

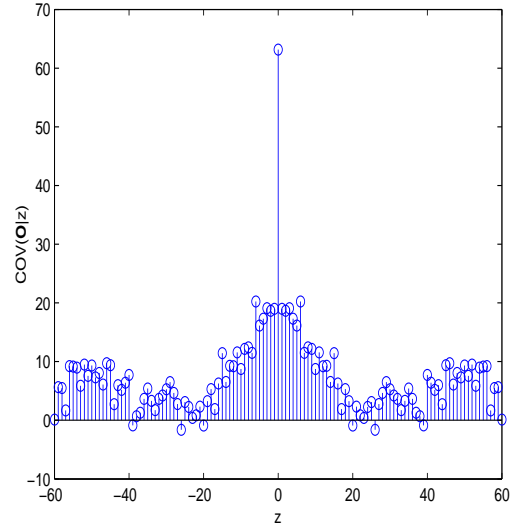


Figure B.15: Measured covariance of the column projections of the image in Figure B.12 with AWGN of $\sigma = 100$.

Bibliography

1. Awate S. P. and Whitaker R. T. "Unsupervised, Information-Theoretic, Adaptive Image Filtering for Image Restoration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(3):364–376 (March 2006).
2. Barash D. "A Fundamental Relationship Between Bilateral Filtering, Adaptive Smoothing and the Nonlinear Diffusion Equation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(6):844–847 (June 2002).
3. Barron J., Fleet D., and Beauchemin S. "Performance of Optical Flow Techniques," *International Journal of Computer Vision*, 12(1):43–77 (1994).
4. Bergen J. R., Anandan P., Hanna K. J., and Hingorani R. "Hierarchical Model-Based Motion Estimation." *ECCV '92: Proceedings of the Second European Conference on Computer Vision*. 237–252. London, UK: Springer-Verlag, 1992.
5. Black M. J. and Sapiro G. "Edges as Outliers: Anisotropic Smoothing Using Local Image Statistics." *SCALE-SPACE '99: Proceedings of the Second International Conference on Scale-Space Theories in Computer Vision*. 259–270. London, UK: Springer-Verlag, 1999.
6. Black M. J., Sapiro G., Marimont D., and Heeger D. "Robust Anisotropic Diffusion," *IEEE Transactions on Image Processing*, 7(3):421–432 (March 1998).
7. Brown L. G. "A Survey of Image Registration Techniques," *ACM Comput. Surv.*, 24(4):325–376 (December 1992).
8. Buades A., Coll B., and Morel J. M. "A Non-Local Algorithm for Image Denoising." *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 60–65. June 2005.
9. Buades A., Coll B., and Morel J. M. "A Review of Image Denoising Algorithms, With a New One," *Multiscale Modelling and Simulation*, 4(2):490–530 (2005).
10. Cain S. C. "Design of an Image Projection Correlating Wavefront Sensor for Adaptive Optics," *Optical Engineering*, 43:1670–1681 (July 2004).
11. Cain S. C., Hayat M. M., and Armstrong E. E. "Projection-Based Image Registration in the Presence of Fixed-Pattern Noise," *IEEE Transactions on Image Processing*, 28(12):1860–1870 (December 2001).
12. Chambolle A. "An Algorithm for Total Variation Minimization and Applications," *Journal of Mathematical Imaging and Vision*, 20:89–97 (2004).
13. Chan T., Osher S., and Shen J. "The Digital TV Filter and Nonlinear Denoising," *IEEE Transactions on Image Processing*, 10(2):231–241 (February 2001).

14. Chang S. G., Yu B., and Vetterli M. "Adaptive Wavelet Thresholding for Image Denoising and Compression," *IEEE Transactions on Image Processing*, 9(9):1532–1546 (September 2000).
15. Chang S. G., Yu B., and Vetterli M. "Spatially Adaptive Wavelet Thresholding with Context Modeling for Image Denoising," *IEEE Transactions on Image Processing*, 9(9):1522–1531 (September 2000).
16. Cheng Y. "Mean Shift, Mode Seeking, and Clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(8):790–799 (August 1995).
17. Dabov K., Foi A., Katkovnik V., and Egiazarian K. "Image Denoising with Block-matching and 3D Filtering." *Proc SPIE: Image Processing: Algorithms and Systems, Neural Networks, and Machine Learning 6064*, edited by E. R. Dougherty, et al. February 2006.
18. Dickinson J. D. and Chakraborti S. *Nonparametric Statistical Inference* (4 Edition), 168. Statistics: Textbooks and Monographs. New York, NY: Marcel Dekker, Inc., 2003.
19. Donoho D. L. and Johnstone I. M. "Adapting to Unknown Smoothness via Wavelet Shrinkage," *Journal of the American Statistical Association*, 90(432):1200–1224 (1995).
20. Driggers R. G., Vollmerhausen R. H., and Krapels K. A. "Target Identification Performance as a Function of Temporal and Fixed Pattern Noise," *Optical Engineering*, 40(3):443–447 (March 2001).
21. Elad M. "On the Origin of the Bilateral Filter and Ways to Improve it," *IEEE Transactions on Image Processing*, 11(10):1141–1151 (October 2002).
22. Elad M., Teo P., and Hel-Or Y. "On the Design of Filters for Gradient-Based Motion Estimation," *Journal of Mathematical Imaging and Vision*, 23(3):345–365 (November 2005).
23. Fukunaga K. and Hostetler L. "The Estimation of the Gradient of a Density Function with Applications in Pattern Recognition," *IEEE Transactions on Information Theory*, 21(1):32–40 (January 1975).
24. Goodman J. W. *Introduction to Fourier Optics* (3 Edition). Greenwood Village, CO: Roberts and Company, 2005.
25. Healey G. and Kondepudy R. "Radiometric CCD Camera Calibration and Noise Estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(3):267–276 (March 1994).
26. Jain A. K. *Fundamentals of Digital Image Processing*. Upper Saddle River, NJ: Prentice-Hall, Inc., 1989.

27. Johnson H. and Christensen G. “Consistent Landmark and Intensity-Based Image Registration,” *IEEE Transactions on Medical Imaging*, 21(5):450–461 (May 2002).
28. Johnson N. L., Kotz S., and Balakrishnan N. *Continuous Univariate Distributions*. New York: Wiley and Sons, 1994.
29. Kay S. M. *Fundamentals of Statistical Signal Processing: Detection Theory*, 2. Englewood Cliffs, N.J.: Prentice-Hall PTR, 1993.
30. Kay S. and Marple, S.L.J. “Spectrum Analysis{ }A Modern Perspective,” *Proceedings of the IEEE*, 69(11):1380–1419 (November 1981).
31. Kervrann C. and Boulanger J. *Local Adaptivity to Variable Smooothness for Exemplar-Based Image Denoising and Representation*. Research Report 1733, Rennes, Fr: Institute de Recherche en Informatique et Systemes Aleatoires, July 2005.
32. Kervrann C. and Boulanger J. “Optimal Spatial Adaptation for Patch-Based Image Denoising,” *IEEE Transactions on Image Processing*, 15(10):2866–2878 (October 2006).
33. Knapp C. and Carter G. “The Generalized Correlation Method for Estimation of Time Delay,” *IEEE Transaction on Acoustics, Speech, and Signal Processing*, 24(4):320–327 (August 1976).
34. Krajsek K. and Mester R. *Lecture Notes in Computer Science - Complex Motion*, chapter Wiener-Optimized Discrete Filters for Differential Motion Estimation, 30–41. Springer Berlin / Heidelberg, 2007.
35. Leon-Garcia A. *Probability and Random Processes for Electrical Engineering* (2 Edition). Reading, MA: Addison Wesley Longman, 1994.
36. MacDonald A. *Blind Deconvolution of Anisoplanatic Images Collected by a Partially Coherent Imaging System*. PhD dissertation, Air Force Institute of Technology, Wright-Patterson AFB, OH, June 2006.
37. McAulay R. and Hofstetter E. “Barankin Bounds on Parameter Estimation,” *IEEE Transactions on Information Theory*, 17(6):669–676 (November 1971).
38. McAulay R. and Seidman L. “A Useful Form of the Barankin Lower Bound and its Application to PPM Threshold Analysis,” *IEEE Transactions on Information Theory*, 15(2):273–279 (March 1969).
39. Nestares O. and Heeger D. “Robust Multiresolution Alignment of MRI Brain Volumes,” *Magnetic Resonance in Medicine*, 43(5):705–715 (May 2000).
40. Perona P. and Malik J. “Scale-Space and Edge Detection Using Anisotropic Diffusion,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629–639 (July 1990).

41. Pinkus A. and Tabrikian J. "Barankin Bound for Range and Doppler Estimation Using Orthogonal Signal Transmission." *2006 IEEE Conference on Radar*. 94–99. April 2006.
42. Pizurica A., Philips W., Lemahieu I., and Acheroy M. "A Joint Inter- and Intrascale Statistical Model for Bayesian Wavelet Based Image Denoising," *IEEE Transactions on Image Processing*, 11(5):545–557 (May 2002).
43. Pluim J., Maintz J., and Viergever M. "Mutual-Information-Based Registration of Medical Images: A Survey," *IEEE Transactions on Medical Imaging*, 22(8):986–1004 (August 2003).
44. Portilla J., Strela V., Wainwright M., and Simoncelli E. "Image Denoising Using Scale Mixtures of Gaussians in the Wavelet Domain," *IEEE Transactions on Image Processing*, 12(11):1338–1351 (November 2003).
45. Robinson D. and Milanfar P. "Efficiency and Accuracy Tradeoffs in using Projections for Motion Estimation." *Signals, Systems and Computers, 2001. Conference Record of the Thirty-Fifth Asilomar Conference on*. 545–550. November 2001.
46. Robinson D. and Milanfar P. "Fundamental Performance Limits in Image Registration," *IEEE Transactions on Image Processing*, 13(9):1185–1199 (September 2004).
47. Robinson D. and Milanfar P. "Bias Minimizing Filter Design for Gradient-Based Image Registration," *Signal Processing: Image Communication*, 20(6):554–568 (July 2005).
48. Robinson D. and Milanfar P. "Statistical performance analysis of super-resolution," *IEEE Transactions on Image Processing*, 15(6):1413–1428 (June 2006).
49. Rudin L., Osher S., and Fatemi E. "Nonlinear Total Variation Based Noise Removal Algorithms," *Physica D*, 60:259–268 (1992).
50. Simoncelli E., Freeman W., Adelson E., and Heeger D. "Shiftable Multiscale Transform," *IEEE Transactions on Information Theory*, 38(2):587–607 (March 1992).
51. Snyder D. L., Hammoud A. M., and White R. L. "Image Recovery from Data Acquired with a Charge-Coupled-Device camera," *Journal of the Optical Society of America. A, Optics and image science*, 10(5):1014–1023 (May 1993).
52. Tomasi C. and Manduchi R. "Bilateral Filtering for Gray and Color Images." *Proceedings of the 1998 IEEE International Conference on Computer Vision, Bombay, India*. 839–846. January 1998.
53. VanTrees H. L. *Detection, Estimation, and Modulation Theory, 1*. New York, NY: John Wiley and Sons, Inc., 2001.

54. Watkins D. S. *Fundamentals of Matrix Computations* (2 Edition). New York, NY: John Wiley and Sons, 2002.
55. Yetik I. and Nehorai A. “Performance Bounds on Image Registration,” *IEEE Transactions on Signal Processing*, 54(5):1737–1749 (May 2006).

REPORT DOCUMENTATION PAGE					<i>Form Approved OMB No. 0704-0188</i>	
<small>The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.</small>						
PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.						
1. REPORT DATE (DD-MM-YYYY)		2. REPORT TYPE			3. DATES COVERED (From - To)	
4. TITLE AND SUBTITLE				5a. CONTRACT NUMBER		
				5b. GRANT NUMBER		
				5c. PROGRAM ELEMENT NUMBER		
6. AUTHOR(S)				5d. PROJECT NUMBER		
				5e. TASK NUMBER		
				5f. WORK UNIT NUMBER		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)					8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)					10. SPONSOR/MONITOR'S ACRONYM(S)	
					11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT						
13. SUPPLEMENTARY NOTES						
14. ABSTRACT						
15. SUBJECT TERMS						
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON	
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPHONE NUMBER (Include area code)	